

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.

PCT

世界知的所有権機関  
国際事務局  
特許協定に基づいて公開された国際出願



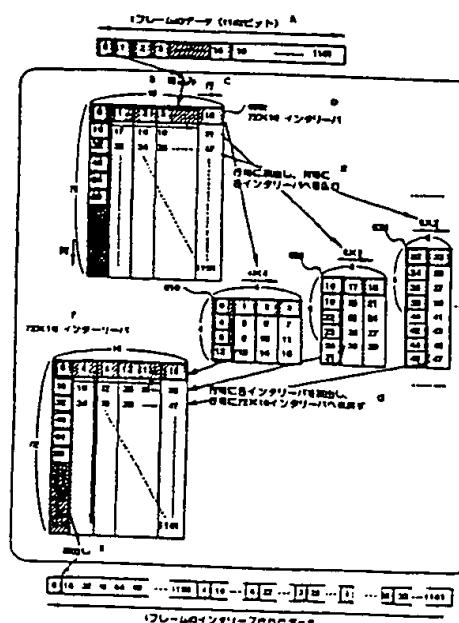
(51) 国際特許分類6 H03M 13/22, 13/12	A1	(11) 国際公開番号 WO99/25069
		(43) 国際公開日 1999年5月20日(20.05.99)
(21) 国際出願番号 PCT/JP98/05027		(74) 代理人 弁理士 伊東忠彦(ITOH, Tadahiko) 〒150-6032 東京都渋谷区恵比寿4丁目20番3号 恵比寿ガーデンプレイスタワー32階 Tokyo, (JP)
(22) 国際出願日 1998年11月9日(09.11.98)		(81) 指定国 CA, CN, JP, KR, US, 欧州特許 (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).
(30) 優先権データ 特願平9/307599 特願平10/218377 特願平10/233088	JP JP JP	(71) 出願人 (米国を除くすべての指定国について) エヌ・ティ・ティ移動通信網株式会社 (NTT MOBILE COMMUNICATIONS NETWORK, INC.)(JP/JP) 〒105-8436 東京都港区虎ノ門二丁目10番1号 Tokyo, (JP)
(72) 発明者: および (73) 発明者/出願人 (米国についてのみ) 渋谷 彰(SHIBUTANI, Akira)[JP/JP] 〒236-0053 神奈川県横浜市金沢区能見台通18-11-303 Kanagawa, (JP) 須田博人(SUDA, Hirohito)[JP/JP] 〒237-0075 神奈川県横須賀市田浦町1-1-5 ファミール湘南田浦1204 Kanagawa, (JP)		(75) 添付公開書類 国際調査報告書

(54) Title: INTERLEAVING METHOD, INTERLEAVING APPARATUS, AND RECORDING MEDIUM IN WHICH INTERLEAVE PATTERN GENERATING PROGRAM IS RECORDED

(54) 発明の名称 インタリービング方法、インターリービング装置、及びインターブパターン作成プログラムを記録した記録媒体

## (57) Abstract

Data of an input data series is written into a first interleaver. The data is read out in units of a column or row from the first interleaver and written into a plurality of second interleavers in the units of a column or row. The data is read from each of the second interleavers and written into one or a plurality of third interleavers as necessary. The operation is repeated once or a plurality of times, thereby reading the data from each of the interleavers and generating a data series. Interleaving is carried out by generating interleaving patterns by using a plurality of interleaving patterns. Further, an interleave pattern adapted to turbo coding or transmission is generated.



A ... DATA CONTAINED IN ONE FRAME (1112 BITS)  
 B ... WRITE  
 C ... READ  
 D ... 72 X 16 INTERLEAVES  
 E ... READ DATA FROM ROW AND WRITE IT EVERY COLUMN TO EACH INTERLEAVER  
 F ... 72 X 16 INTERLEAVER  
 G ... READ DATA FROM EACH INTERLEAVER EVERY COLUMN AND WRITE IT EVERY ROW TO 72 X 16 INTERLEAVER  
 H ... READ  
 I ... INTERLEAVED DATA CONTAINED IN ONE FRAME

入力データ系列のデータを第1のインタリーバに書き込み、該第1のインタリーバから列又は行単位にデータを読み出し、該列又は行単位毎に、データを複数の第2のインタリーバに書き込み、該第2のインタリーバの各々からデータを読み出し、該データを必要に応じて1又は複数の第3のインタリーバに書き込むことを1回又は複数回繰り返した結果のインタリーバの各々からデータを読み出してデータ系列を出力する。また、インターブパターンを複数用いてインターブパターンを作成し、インターブを行う。更に、ターボ符号化又は伝送等に適したインターブパターンを作成する。

PCTに基づいて公開される国際出願のパンフレット第一頁に掲載されたPCT加盟国を同定するために使用されるコード(参考情報)

AE	アラブ首長国連邦	ES	スペイン	LI	リヒテンシュタイン	SG	シンガポール
AL	アルバニア	FI	フィンランド	LK	スリ・ランカ	SI	スロヴェニア
AM	アルメニア	FR	フランス	LR	リベリア	SK	スロヴァキア
AT	オーストリア	GA	ガボン	LS	レソト	SL	シエラ・レオネ
AU	オーストラリア	GB	英國	LT	リトアニア	SN	セネガル
AZ	アゼルバイジャン	GD	グレナダ	LU	ルクセンブルグ	SZ	スウェーデン
BA	ボスニア・ヘルツェゴビナ	GE	グルジア	LV	ラトヴィア	TD	チャード
BB	ベルバドス	GH	ガーナ	MC	モナコ	TG	トーゴー
BE	ベルギー	GM	ガンビア	MD	モルドヴァ	TJ	タジキスタン
BF	ブルガリア	GN	ギニア	MG	マダガスカル	TM	トルクメニスタン
BC	ブルガリア	GW	ギニア・ビサオ	MK	マケドニア旧ユーコスラヴィア	TR	トルコ
BJ	ベナン	GR	ギリシャ	NP	共和国	TT	トリニダッド・トバゴ
BR	ブラジル	HR	クロアチア	ML	マリ	UA	ウクライナ
BY	ベラルーシ	HU	ハンガリー	MN	モンゴル	UG	ウガンダ
CA	カナダ	ID	インドネシア	MR	モーリタニア	US	米国
CF	中央アフリカ	IE	アイルランド	MW	マラウイ	UZ	ウズベキスタン
CG	コンゴー	IL	イスラエル	MX	メキシコ	VN	ヴィエトナム
CH	スイス	IN	インド	NE	ニジェール	YU	ユーゴースラビア
CI	コートジボアール	IS	アイスランド	NL	オランダ	ZA	南アフリカ共和国
CM	カメルーン	IT	イタリア	NO	オランダ	ZW	ジンバブエ
CN	中国	JP	日本	NZ	ニュー・ジーランド		
CU	キューバ	KE	ケニア	PL	ポーランド		
CY	キプロス	KC	キルギスタン	PT	ポルトガル		
CZ	チェコ	KP	北朝鮮	RO	ルーマニア		
DE	ドイツ	KR	韓国	RU	ロシア		
DK	デンマーク	KZ	カザフスタン	SD	スードン		
EE	エストニア	LC	セントルシア	SE	スウェーデン		

(19)日本国特許庁 (JP)

# 再公表特許 (A1)

(11)国際公開番号

WO 99/25069

発行日 平成12年2月29日 (2000.2.29)

(43)国際公開日 平成11年5月20日 (1999.5.20)

(51)Int.Cl.<sup>7</sup>

H 03 M 13/22  
13/12

識別記号

F I

審査請求 有 予備審査請求 未請求(全101頁)

出願番号 特願平11-525965  
(21)国際出願番号 PCT/JP98/05027  
(22)国際出願日 平成10年11月9日 (1998.11.9)  
(31)優先権主張番号 特願平9-307599  
(32)優先日 平成9年11月10日 (1997.11.10)  
(33)優先権主張国 日本 (JP)  
(31)優先権主張番号 特願平10-218377  
(32)優先日 平成10年7月31日 (1998.7.31)  
(33)優先権主張国 日本 (JP)  
(31)優先権主張番号 特願平10-233088  
(32)優先日 平成10年8月19日 (1998.8.19)  
(33)優先権主張国 日本 (JP)

(71)出願人 エヌ・ティ・ティ移動通信網株式会社  
東京都港区虎ノ門2丁目10番1号  
(72)発明者 渋谷 彰  
神奈川県横浜市金沢区能見台通18-11-  
303  
(72)発明者 須田 博人  
神奈川県横須賀市田浦町1-1-5 ファ  
ミール湘南田浦1204  
(74)代理人 弁理士 伊東 忠彦

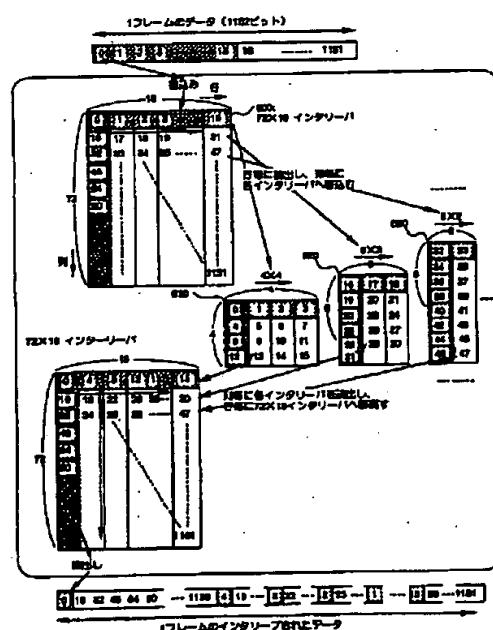
最終頁に続く

(54)【発明の名称】 インタリーピング方法、インタリーピング装置、及びインタリーブパターン作成プログラムを記録した記録媒体

(57)【要約】

入力データ系列のデータを第1のインタリーバに書き込み、該第1のインタリーバから列又は行単位にデータを読み出し、該列又は行単位毎に、データを複数の第2のインタリーバに書き込み、該第2のインタリーバの各々からデータを読み出し、該データを必要に応じて1又は複数の第3のインタリーバに書き込むことを1回又は複数回繰り返した結果のインタリーバの各々からデータを読み出してデータ系列を出力する。また、インタリーブパターンを複数用いてインタリーブパターンを作成し、インタリーブを行う。更に、ターボ符号化又は伝送等に適したインタリーブパターンを作成する。

FIG. 6



**【特許請求の範囲】**

1. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、

データ系列のデータを第1のインタリーバに書き込み、該第1のインタリーバから列又は行単位にデータを読み出し、該列又は行単位毎に、データを複数の第2のインタリーバに書き込む第1のステップを実行し、

該第2のインタリーバの各々からデータを読み出し、該データを必要に応じて1又は複数の第3のインタリーバに書き込む第2のステップを1回又は複数回繰り返した結果のインタリーバの各々から、又は第1のステップの結果のインタリーバの各々からデータを読み出してデータ系列を出力することを特徴とするインタリービング方法。

2. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、

データ系列を第1のインタリーバに対して一方向に書き込む第1のステップを実行し、

該第1のインタリーバからデータを列又は行単位に読み出し、該第1のインタリーバと異なる容量の第2のインタリーバに該読み出したデータを一方向に書き込むことを該列又は行単位に繰り返す第2のステップを実行し、

第2のステップにより生成された複数個の該第2のインタリーバの各々を前記第1のインタリーバとみなして前記第2のステップを各々のインタリーバ毎に実行する第3のステップを繰り返し、

第3のステップの繰り返しから生成又は第2のステップから生成されたインタリーバの各々からデータを読み出してデータ系列を出力することを特徴とするインタリービング方法。

3. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、

データ系列を第1のインタリーバに対して一方向に書き込む第1のステップを実行し、

該第1のインタリーバからデータを列又は行単位に読み出し、該第1のインタリーバと異なる容量の第2のインタリーバに該読み出したデータを一方向に書き込むことを該列又は行単位に繰り返す第2のステップを実行し、

該第2のステップの結果から生成したインタリーバの各々から列又は行単位にデータを読み出し、該データを前記第1のインタリーバと同一容量のインタリーバに書き込む第3のステップを実行することによりにより生成されたインタリーバからデータを読み出してデータ系列を出力することを特徴とするインタリービング方法。

4. 請求項3記載のインタリービング方法において、

前記第3のステップを実行することにより生成されたインタリーバを前記第1のインタリーバとみなして前記第2のステップ及び第3のステップを実行する第4のステップを1回または複数回繰り返すことにより生成されたインタリーバからデータを読み出してデータ系列を出力することを特徴とするインタリービング方法。

5. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、

複数のインタリービングパターンを予めテーブルに登録しておき、該テーブルを参照して、入力したデータ系列に該複数のインタリービングパターンのいずれかを適用して出力し、該出力に対してさらに前記複数のインタリービングパターンのいずれかを適用して出力することを繰り返すことを特徴とするインタリービング方法。

6. 請求項5記載のインタリービング方法において、

前記テーブルは、少なくとも請求項1から4のうちいずれか記載のインタリービング方法によるインタリービングパターンを予め登録しておくことを特徴とするインタリービング方法。

7. ある単位長のデータ系列を入力して該単位長のインタリープ

した系列を出力するインタリービング方法においてインタリービングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作

成方法において、

第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を用いて、第3の単位のインタリープパターン記述を作成することを特徴とするインタリープパターン記述作成方法。

8. 請求項7記載のインタリープパターン記述作成方法を複数回用いることで、所定長単位のインタリープパターン記述を作成することを特徴とするインタリープパターン記述作成方法。

9. 請求項7または8記載のインタリープパターン記述作成方法において、

前記インタリープパターン記述は、インタリープパターンを記述したインタリープパターンテーブルまたはインタリープパターン方程式であることを特徴とするインタリープパターン記述作成方法。

10. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、

第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を用いて、第3の単位のインタリープパターン記述を作成し、作成したインタリープパターン記述を用いてインタリービング処理を行うことを特徴とするインタリービング方法。

11. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、

第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を用いて、第3の単位の系列におけるインタリービング先を計算し、

該計算結果に基づきインタリービング処理を行うことを特徴とするインタリービング方法。

12. 請求項7ないし8いずれか記載のインタリープパターン記

述作成方法で作成されたインタリープパターン記述を用いて、第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を作成し、

作成された該第1の単位のインタリープパターン記述と該第2の単位のインタリープパターン記述から計算することにより、第3の単位のデータ系列に対して

インタリービング処理を行うことを特徴とするインタリービング方法。

13. 請求項10ないし12いずれか記載のインタリービング方法において、

前記インタリーブパターン記述は、インタリーブパターンを記述したインタリーブパターンテーブルまたはインタリーブパターン方程式であることを特徴とするインタリービング方法。

14. ある単位長のデータ系列を入力して該単位長のインタリーブした系列を出力するインタリービング方法においてインタリービングパターンを使用して入力データをインタリーブする場合の該インタリーブパターンの記述作成方法において、

インタリーブパターンを定義するインタリーブパターン記述言語を解釈し、解釈した結果に基づき、請求項9記載のインタリーブパターン記述作成方法を用いて、インタリーブパターン記述を作成することを特徴とするインタリーブパターン記述作成方法。

15. 請求項14記載のインタリーブパターン記述作成方法において、

インタリーブパターンを作成するとき、インタリーブパターン記述言語の一部に対応するインタリーブパターン記述をすでに保持していた場合、その記述言語の一部に対応する処理を行わずに、保持しているインタリーブパターン記述を参照してインタリーブパターン記述を作成することを特徴とするインタリーブパターン記述作成方法。

16. インタリーブパターンを定義するインタリーブパターン記述言語を解釈し、

解釈した結果に基づき、請求項13記載のインタリービング方法を用いてインタリービングを行うことを特徴とするインタリービング方法。

17. 請求項16記載のインタリービング方法において、

インタリービングを行う際、インタリーブパターン記述言語の一部に対応するインタリーブパターン記述をすでに保持していた場合、その記述言語の一部に対応する処理を行わずに、保持しているインタリーブパターンを参照してインタリーブパターン記述を作成することを特徴とするインタリービング方法。

18. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法においてインタリーピングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、

ある単位長が与えられたとき、まず、一段目のインタリープパターン記述を決定し、

次に、一段目以降の縦および横それぞれのインタリーバに対応するインタリープパターン記述を決定することを、任意の段またはインタリーピングができなくなるまで繰り返すことにより、インタリープパターン記述を生成することを特徴とするインタリープパターン記述作成方法。

19. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法においてインタリーピングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、

生成されたインタリープパターン記述を検査し、検査に不合格であれば、パラメータの一部もしくは全てを変更し、インタリープパターン記述を再生成し、検査に合格するまでこれを繰り返し、最終

的に、検査に合格したインタリープパターン記述を生成することを特徴とするインタリープパターン記述作成方法。

20. 請求項18または19記載のインタリープパターン記述作成方法において、前記生成されるインタリープパターン記述は、インタリープパターンテーブル、インタリープパターン方程式、またはインタリープパターン記述言語であることを特徴とするインタリープパターン記述作成方法。

21. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法においてインタリーピングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、

ある単位長が与えられたとき、該単位長に対応するブロックインタリーバの行

数または列数を予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定した行数または列数から列数または行数を定めるステップを、該定められた列数または行数が該インタリープ・パターン・リストに定義されるまで繰り返し実行して得られたインタリープ・パターンから前記単位長のインタリープ・パターンを作成することを特徴とするインタリープパターン作成方法。

22. 請求項21記載のインタリープパターン作成方法において、

ある単位長が与えられたとき、まず、第1段目の処理として、定められた数により、該単位長に対応するブロックインタリーバの行数または列数を決定し、その数に対応するインタリープパターンを予め定めたインタリープパターンとし、該決定された列数を用いて行数を定め、または、該決定された行数を用いて列数を定める第1のステップを実行し、

前記定められた行数または列数に対応するブロックインタリーバの行数または列数を、予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定された行数から列

数を定め、または、決定された列数から行数を定める第2のステップを、行数または列数に対応するインタリープパターンが前記予め定められているインタリープ・パターン・リスト中に存在するまで繰り返す第3のステップを実行し、

該第3のステップを前記第1のステップにおける前記予め定めたインタリープパターンに対応する行数回または列数回行い、

得られた最終段階の行と列に対応するインタリープ・パターンから、順次前の段階の行または列に対応するインタリープ・パターンを作成し、結果として前記単位長のインタリープパターンを作成する

ことを特徴とするインタリープパターン作成方法。

23. 請求項21または22記載のインタリープパターン作成方法において、更に、作成した前記単位長のインタリープパターンをチェックし、該チェック結果により、再度該単位長のインタリープパターンを作成し直すことを特徴とするインタリープパターン作成方法。

24. 請求項22または23記載のインタリープパターン作成方法において、適応対象としてターボ符号化であり、第1段目の行数を7とすることを特徴とするインタリープパターン作成方法。

25. 請求項22または23記載のインタリープパターン作成方法において、適応対象として伝送であり、第1段目の列数を1フレームのスロット数とすることを特徴とするインタリープパターン作成方法。

26. ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービングを実行するインタリービング装置において、

1または複数のインタリービングパターンを予めテーブルに登録しておく手段と、

該テーブルを参照して、入力したデータ系列に該複数のインタ

リービングパターンのいずれかを適用して出力する手段と、

必要に応じて該出力に対してさらに該複数のインタリービングパターンのいずれかを適用して出力することを繰り返す手段と  
を有することを特徴とするインタリービング装置。

27. 請求項26記載のインタリービング装置において、  
前記テーブルは、少なくとも請求項1から4のうちいずれか記載のインタリービング方法によるインタリービングパターンを予め登録しておくことを特徴とする  
インタリービング装置。

28. 請求項26記載のインタリービング装置において、

前記インタリービングパターンは、請求項21記載のインタリービングパターン作成方法によることを特徴とするインタリービング装置。

29. 請求項26ないし28のうちいずれか1項に記載のインタリービング装置において、

インタリービングパターンを使用する代りに、入力したデータ系列のインタリービング先を計算し、該計算結果に基づきインタリービング処理を行いデータ出力することを特徴とするインタリービング装置。

30. ある単位長のデータ系列を入力して該単位長のインタリープした系列を

出力するインタリーピング方法におけるインタリープバターンの記述作成を行うプログラムを記録したコンピュータ読み取り可能な記録媒体において、該プログラムは、

ある単位長が与えられたとき、該単位長に対応するブロックインタリーバの行数または列数を予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定した行数または列数から列数または行数を定めるステップを、該定められた列数または行数が該インタリープ・パターン・リストに定義されるまで繰り返し実行して得られたインタリープ・パターンから前記単位長のインタリープ・パターンを作成することを特徴とするインタ

リープパターン作成プログラムを記録した記録媒体。

31. 請求項30記載のインタリープパターン作成プログラムを記録した記録媒体において、該プログラムは、

ある単位長が与えられたとき、まず、第1段目の処理として、定められた数により、該単位長に対応するブロックインタリーバの行数または列数を決定し、その数に対応するインタリープパターンを予め定めたインタリープパターンとし、該決定された列数を用いて行数を定め、または、該決定された行数を用いて列数を定める第1のステップを実行し、

前記定められた行数または列数に対応するブロックインタリーバの行数または列数を、予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定された行数から列数を定め、または、決定された列数から行数を定める第2のステップを、行数または列数に対応するインタリープパターンが前記予め定められているインタリープ・パターン・リスト中に存在するまで繰り返す第3のステップを実行し、

該第3のステップを前記第1のステップにおける前記予め定めたインタリープパターンに対応する行数回または列数回行い、

得られた最終段階の行と列に対応するインタリープ・パターンから、順次前の段階の行または列に対応するインタリープ・パターンを作成し、結果として前記単位長のインタリープパターンを作成する

ことを特徴とするインタリープパターン作成プログラムを記録した記録媒体。

32. 請求項30または31記載のインタリープパターン作成プログラムを記録した記録媒体において、前記プログラムは、更に、作成した前記単位長のインタリープパターンをチェックし、該チェック結果により、再度該単位長のインタリープパターンを作成し直すことを特徴とするインタリープパターン作成プログラムを記

録した記録媒体。

33. 請求項31記載のインタリープパターン作成プログラムを記録した記録媒体において、前記プログラムは、適応対象としてターボ符号化であり、第1段目の行数を7とすることを特徴とするインタリープパターン作成プログラムを記録した記録媒体。

34. 請求項31記載のインタリープパターン作成プログラムを記録した記録媒体において、前記プログラムは、適応対象として伝送であり、第1段目の列数を1フレームのスロット数とすることを特徴とするインタリープパターン作成プログラムを記録した記録媒体。

## 【発明の詳細な説明】

インタリービング方法、インタリービング装置、及び  
インタリープパターン作成プログラムを記録した記録媒体

### 技術分野

本発明は、バースト誤りに対する誤り訂正符号の能力向上等のためのインタリービング技術に係り、特に、データのランダム性を増加させてインタリービングの効果を向上させるインタリービング方法、インタリービング装置、及びインタリープパターン作成プログラムを記録した記録媒体に関する。

### 背景技術

移動通信システム等のデジタル伝送では、建物等の反射によるマルチパス・フェージングによって受信信号のレベルは時間的に大きく変動し、それによりバースト誤り等の符号誤りが生じる。また、CDやハードディスクなどのデジタル・システムの記録媒体において、傷や読み取り面の埃などによってバースト誤り等の符号誤りが生じる。このため、システムにおいて、各種誤り訂正符号を使用することになるが、この様な誤り訂正符号において、バースト誤りに対する訂正能力を向上するために、インタリービング技術が組み合わせて用いられる。このインタリービング技術の良し悪しが、バースト誤り存在下の誤り訂正符号の能力を決定する。

また、近年提案された能力の高い誤り訂正符号を用いるターボ符号器は複数の符号器で構成されており、各符号器間の冗長系列の相関性を少なくするためにインタリーバ（インタリービング処理を行う手段）を介して各符号器間が連接されている。このインタリーバは、ターボ符号の能力を決定する大変重要なものとなっている。

したがって、このようなインタリーバを用いているターボ符号化

や上記の移動通信システム等の伝送系に適したインタリービング処理を実行する方法が求められている。

当業者には公知の如く、インタリービング（interleaving）方法は、入力ビット系列のビットの順番と出力ビット系列のビットの順番とをランダ

ム化することを目的としており、以下の観点は、インタリービング法の能力を評価する尺度として用いることができる。

- (1) 連続する2ビットの入力を、出力系列においてどれだけ遠くに離せるか。
- (2) 連続する2ビットの出力は、入力系列においてどれだけ遠くに離れているか。

図1は、従来のインタリービング方法であるブロック・インタリービング方法を示す。

図1において、1フレームのデータ100は1152ビットで構成されている。配列110は、 $N \times M$  ( $N$ 行 $M$ 列) のバッファを持ち、このバッファに例えば斜線部Aの行ベクトル115のように行方向に $M$ ビット書き込み、斜線部Bの列ベクトル120のように列方向に $N$ ビット読み出すことでインタリービング方法を実現している。このインタリービング方法について上記の観点で評価すると、

- (1) 連続する2ビットの入力は、インタリーブ後の出力系列130において $N$ ビットよりも遠くに引き離せず、
- (2) 連続する2ビットの出力は、入力系列においては少なくとも $M$ ビット離れている。

しかしながら、上述のインタリービング方法では、行方向に書き込むときには入力したビット系列における時間順に書いており、列方向に読み出すときにも入力したビット系列における時間順に読み出しているので、ともに1回づつ時間順に読み書きしている処理で留まっており、インタリービングの効果が低く、 $N \times M$ のバッファを持ちなながら上記程度のランダマイズの能力に留まっていた。

#### 発明の開示

本発明は、上記の点に鑑みなされたものであり、バッファに対する書き込みまたは読み出しの処理を1回行った後、さらに順番を入れ替える処理を繰り返して適用することにより、1回づつ時間順に読み書きしている場合よりもインタリービングの効果を向上させるインタリービング方法を提供することを第1の目的とする。

上記の目的を達成するため、本発明によるインタリービング方法は、ある単位

長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法において、データ系列のデータを第1のインタリーバに書き込み、該第1のインタリーバから列又は行単位にデータを読み出し、該列又は行単位毎に、データを複数の第2のインタリーバに書き込む第1のステップを実行し、

該第2のインタリーバの各々からデータを読み出し、該データを必要に応じて1又は複数の第3のインタリーバに書き込む第2のステップを1回又は複数回繰り返した結果のインタリーバの各々から、又は第1のステップの結果のインタリーバの各々からデータを読み出してデータ系列を出力する。

上記発明を次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法において、データ系列を第1のインタリーバに対して一方向に書き込む第1のステップを実行し、該第1のインタリーバからデータを列又は行単位に読み出し、該第1のインタリーバと異なる容量の第2のインタリーバに該読み出したデータを一方向に書き込むことを該列又は行単位に繰り返す第2のステップを実行し、第2のステップにより生成された複数個の該第2のインタリーバの各々を前記第1のインタリーバとみなして前記第2のステップを各々のインタリーバ毎に実行する第3のステップを繰り返し、第3のステップの繰り返しから生成又は第2のステップから生成されたインタリーバの各々からデータを読み出してデータ系列を出力する。

上記インタリーピング方法を次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法において、データ系列を第1のインタリーバに対して一方向に書き込む第1のステップを実行し、該第1のインタリーバからデータを列又は行単位に読み出し、該第1のインタリーバと異なる容量の第2のインタリーバに該読み出したデータを一方向に書き込むことを該列又は行単位に繰り返す第2のステップを実行し、該第2のステップの結果から生成したインタリーバの各々から列又は行単位にデータを読み出し、該データを前記第1のインタリーバと同一容量のインタリーバに書き込む第3のステップを実行することによりにより生成されたインタリーバからデータを読み出してデータ系列を出力する。

上記構成において、前記第3のステップを実行することにより生成されたインタリーバを前記第1のインタリーバとみなして前記第2のステップ及び第3のステップを実行する第4のステップを1回または複数回繰り返すことにより生成されたインタリーバからデータを読み出してデータ系列を出力する。

上記インタリービング方法はまた次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、複数のインタリービングパターンを予めテーブルに登録しておき、該テーブルを参照して、入力したデータ系列に該複数のインタリービングパターンのいずれかを適用して出力し、該出力に対してさらに前記複数のインタリービングパターンのいずれかを適用して出力する。

上記構成において、前記テーブルは、少なくとも請求項1から4のうちいずれか記載のインタリービング方法によるインタリービングパターンを予め登録しておく。

上記の発明によれば、バッファに対する書き込みまたは読み出しの処理を1回行った後、さらに順番を入れ替える処理を繰り返して

適用するので、1回づつ時間順に読み書きしている場合よりもインタリービングの効果を向上させることが可能となる。

また、本発明は、インタリービングに要するデータ量を低減しながら種々のインタリービングに柔軟に対応することができるインタリービング方法を提供することを第2の目的とする。

上記の目的を達成するために、本発明は、ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法においてインタリービングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を用いて、第3の単位のインタリープパターン記述を作成するインタリープパターン記述作成方法である。

本発明はまた、上記構成におけるインタリープパターン記述作成方法を複数回用いることで、所定長単位のインタリープパターン記述を作成する。

上記構成において、前記インタリープパターン記述は、インタリープパターンを記述したインタリープパターンテーブルまたはインタリープパターン方程式である。

本発明のインタリービング方法は次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を用いて、第3の単位のインタリープパターン記述を作成し、作成したインタリープパターン記述を用いてインタリービング処理を行う。

本発明のインタリービング方法はまた次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法において、第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を用

いて、第3の単位の系列におけるインタリービング先を計算し、該計算結果に基づきインタリービング処理を行う。

本発明はまた、上記構成のインタリープパターン記述作成方法で作成されたインタリープパターン記述を用いて、第1の単位のインタリープパターン記述と第2の単位のインタリープパターン記述を作成し、作成された該第1の単位のインタリープパターン記述と該第2の単位のインタリープパターン記述から計算することにより、第3の単位のデータ系列に対してインタリービング処理を行う。

本発明はまた、上記インタリービング方法において、前記インタリープパターン記述は、インタリープパターンを記述したインタリープパターンテーブルまたはインタリープパターン方程式である。

上記本発明によれば、インタリープ・パターン記述を作成することにより、例えば、一つのインタリープ・パターン・テーブル（ $A = B$  の場合）、または2つのインタリープ・パターン・テーブルから、よりインタリービング長の大きいインタリープ・パターン・テーブルを作成することが可能である。これにより、あるインタリープ長のパターンをより小さいインタリープ長のパターンの複数の組み合わせで表わすことができ、固定長パターンのメモリ量を削減できる。例えば

、1000ビット分のインタリープ・パターン・テーブルは従来だと1000ビット分のメモリが必要だが、本発明を用いると1000ビット・インタリープ・パターン・テーブルを20ビット・インタリープ・パターン・テーブル×50ビット・インタリープ・パターン・テーブルで表現することで70 (= 20 + 50) ビット分のメモリに削減することができる。また、900ビット・インタリープ・パターン・テーブルを20ビット・インタリープ・パターン・テーブル×50ビット・インタリープ・パターン・テーブルで表現することで、固定長のインタリープ・パターン・テーブルを増加させずに、1000ビットと900ビットのインタリーピングを行うことができる。

本発明のインタリープパターンの記述作成方法は次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリーピング方法においてインタリーピングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、インタリープパターンを定義するインタリープパターン記述言語を解釈し、解釈した結果に基づき、請求項9記載のインタリープパターン記述作成方法を用いて、インタリープパターン記述を作成する。

また、上記構成において、インタリープパターンを作成するとき、インタリープパターン記述言語の一部に対応するインタリープパターン記述をすでに保持していた場合、その記述言語の一部に対応する処理を行わずに、保持しているインタリープパターン記述を参照してインタリープパターン記述を作成する。

また、本発明は、インタリープパターンを定義するインタリープパターン記述言語を解釈し、解釈した結果に基づき、請求項13記載のインタリーピング方法を用いてインタリーピングを行う。

上記構成のインタリーピング方法において、インタリーピングを行う際、インタリープパターン記述言語の一部に対応するインタリープパターン記述をすでに保持していた場合、その記述言語の一部に対応する処理を行わずに、保持しているインタリープパターンを参照してインタリープパターン記述を作成する。

本発明のインタリープパターンの記述作成方法は次のように構成しても良い。

ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法においてインタリービングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、ある単位長が与えられたとき、まず、一段目のインタリープパターン記述を決定し、次に、一段目以降の縦および横それぞれのインタリーパーに対応するインタリープパターン記述を決定することを、任意の段または

インタリービングができなくなるまで繰り返すことにより、インタリープパターン記述を生成する。

また、本発明のインタリープパターンの記述作成方法は次のように構成しても良い。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法においてインタリービングパターンを使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、生成されたインタリープパターン記述を検査し、検査に不合格であれば、パラメータの一部もしくは全てを変更し、インタリープパターン記述を再生成し、検査に合格するまでこれを繰り返し、最終的に、検査に合格したインタリープパターン記述を生成する。

上記構成のインタリープパターン記述作成方法において、前記生成されるインタリープパターン記述は、インタリープパターンテーブル、インタリープパターン方程式、またはインタリープパターン記述言語である。

上記の発明により、インタリービングに要するデータ量を低減しながら種々のインタリービングに柔軟に対応することができる。

また、本発明は、インタリービング方法を伝送系装置やターボ符号化器等に適用する方法、その伝送系装置や符号化器等の装置、及びある対象に適したインタリービングパターン作成プログラムを記録した記録媒体を提供することを第3の目的とする。

上記の目的を達成するために、本発明のインタリープパターン作成方法は次のように構成される。ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法においてインタリービングパターンを

使用して入力データをインタリープする場合の該インタリープパターンの記述作成方法において、ある単位長が与えられたとき、該単位長に対応するブロックインタリーバの行数または列数を予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定した行数

または列数から列数または行数を定めるステップを、該定められた列数または行数が該インタリープ・パターン・リストに定義されるまで繰り返し実行して得られたインタリープ・パターンから前記単位長のインタリープ・パターンを作成する。

また、上記構成において、ある単位長が与えられたとき、まず、第1段目の処理として、定められた数により、該単位長に対応するブロックインタリーバの行数または列数を決定し、その数に対応するインタリープパターンを予め定めたインタリープパターンとし、該決定された列数を用いて行数を定め、または、該決定された行数を用いて列数を定める第1のステップを実行し、前記定められた行数または列数に対応するブロックインタリーバの行数または列数を、予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定された行数から列数を定め、または、決定された列数から行数を定める第2のステップを、行数または列数に対応するインタリープパターンが前記予め定められているインタリープ・パターン・リスト中に存在するまで繰り返す第3のステップを実行し、該第3のステップを前記第1のステップにおける前記予め定めたインタリープパターンに対応する行数回または列数回行い、得られた最終段階の行と列に対応するインタリープ・パターンから、順次前の段階の行または列に対応するインタリープ・パターンを作成し、結果として前記単位長のインタリープパターンを作成する。

また、本発明は上記構成において、更に、作成した前記単位長のインタリープパターンをチェックし、該チェック結果により、再度該単位長のインタリープパターンを作成し直す。

また、上記構成において、適応対象としてターボ符号化であり、第1段目の行数を7とする。

更に、適応対象として伝送であり、第1段目の列数を1フレームのスロット数とする。

上記の発明により、ターボ符号化や伝送等に適したインタリープパターンを得ることができる。

上記目的を達成するための本発明の装置は、ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービングを実行するインタリービング装置において、1または複数のインタリービングパターンを予めテーブルに登録しておく手段と、該テーブルを参照して、入力したデータ系列に該複数のインタリービングパターンのいずれかを適用して出力する手段と、必要に応じて該出力に対してさらに該複数のインタリービングパターンのいずれかを適用して出力することを繰り返す手段とを有する。

また、上記構成において、前記テーブルは、少なくとも請求項1から4のうちいずれか記載のインタリービング方法によるインタリービングパターンを予め登録しておく。

また、上記構成において、前記インタリービングパターンは、請求項21記載のインタリービングパターン作成方法による。

更に、上記インタリービング装置において、インタリービングパターンを使用する代りに、入力したデータ系列のインタリービング先を計算し、該計算結果に基づきインタリービング処理を行いデータ出力する。

上記の発明により、インタリービング処理を行う装置を提供することができ、特に、ターボ符号化や伝送等に適した装置を提供できる。

上記目的を達成するための本発明によるプログラムを記録した記録媒体は、ある単位長のデータ系列を入力して該単位長のインタリープした系列を出力するインタリービング方法におけるインタリープパターンの記述作成を行うプログラムを記録したコンピュータ読み取り可能な記録媒体において、該プログラムは、ある単位長が与えられたとき、該単位長に対応するロックインタリーバの行数または列数を予め定められている適用対象に適したインタリープ

・パターン・リストを用いて決定し、決定した行数または列数から列数または行数を定めるステップを、該定められた列数または行数が該インタリープ・パターン・リストに定義されるまで繰り返し実行して得られたインタリープ・パターンから前記単位長のインタリープ・パターンを作成する。

更に、上記構成において、本発明のインタリープパターン作成プログラムを記録した記録媒体において、該プログラムは、ある単位長が与えられたとき、まず、第1段目の処理として、定められた数により、該単位長に対応するブロックインタリーバの行数または列数を決定し、その数に対応するインタリープパターンを予め定めたインタリープパターンとし、該決定された列数を用いて行数を定め、または、該決定された行数を用いて列数を定める第1のステップを実行し、前記定められた行数または列数に対応するブロックインタリーバの行数または列数を、予め定められている適用対象に適したインタリープ・パターン・リストを用いて決定し、決定された行数から列数を定め、または、決定された列数から行数を定める第2のステップを、行数または列数に対応するインタリープパターンが前記予め定められているインタリープ・パターン・リスト中に存在するまで繰り返す第3のステップを実行し、該第3のステップを前記第1のステップにおける前記予め定めたインタリープパターンに対応する行数回または列数回行い、得られた最終段階の行と列に対応するインタリープ・パターンから、順次前の段階の行または列に対応するインタリープ・パターンを作成し、結果として前記単位長のインタリープパターンを作成する。

上記構成において、本発明のインタリープパターン作成プログラムを記録した記録媒体において、前記プログラムは、更に、作成した前記単位長のインタリープパターンをチェックし、該チェック結果により、再度該単位長のインタリープパターンを作成し直す。

更に、本発明のインタリープパターン作成プログラムを記録した記録媒体において、前記プログラムは、適応対象としてターボ符号化であり、第1段目の行数を7とする。

更に、本発明のインタリープパターン作成プログラムを記録した記録媒体にお

いて、前記プログラムは、適応対象として伝送であり、第1段目の列数を1フレームのスロット数とする。

上記の発明により、インタリービングパターン作成を行うプログラムを記録した記録媒体を提供することができ、特に、ターボ符号化や伝送等に適したインタリービングパターン作成を行うプログラムを記録した記録媒体を提供することができる。

#### 図面の簡単な説明

図1は、従来のインタリービング方法を示す図である。

図2は、本発明の第1の実施の形態のインタリービング方法を示す図である。

図3は、本発明の第1の実施の形態のインタリービング方法を示す図である。

図4は、本発明の第2の実施の形態のインタリービング方法を示す図である。

図5は、本発明の第3の実施の形態のインタリービング方法を示す図である。

図6は、本発明の第4の実施の形態のインタリービング方法を示す図である。

図7は、本発明の第5の実施の形態のインタリービング方法を示す図である。

図8は、 $L = 64$ ビット長のデータが8ビット×8ビットのブロックインタリーバに書き込まれたことを示す図である。

図9は、シンボル単位のインタリービングにおける第1の場合を示す図である。

。

図10は、シンボル単位のインタリービングにおける第2の場合

を示す図である。

図11は、シンボル単位のインタリービングにおける第3の場合を示す図である。

図12は、インタリープ処理を説明する図である。

図13は、本発明の第7の実施の形態のインタリープ処理を説明する図である。

。

図14は、本発明の第7の実施の形態のインタリープ処理を説明する図である。

。

図15は、本発明の第7の実施の形態のインタリープ処理を説明する図である。

- 図16は、本発明の第8の実施の形態のインタリープ処理を説明する図である
- 図17は、本発明の第8の実施の形態のインタリープ処理を説明する図である
- 図18は、本発明の第9の実施の形態のインタリープ処理を説明する図である
- 図19は、本発明の第9の実施の形態のインタリープ処理を説明する図である
- 図20は、本発明の第9の実施の形態のインタリープ処理を説明する図である
- 図21は、本発明の第9の実施の形態のインタリープ処理を説明する図である
- 図22は、本発明の第9の実施の形態のインタリープ処理を説明する図である
- 図23は、インタリープ・パターン記述言語の定義の例を説明する図である。
- 図24は、インタリープ・パターン記述言語の定義の例を説明する図である。
- 図25は、インタリープ・パターン記述言語の定義の例を説明す

る図である。

- 図26は、インタリープ・パターン記述言語の定義の例を説明する図である。
- 図27は、インタリープ・パターン記述言語で定義された場合の実現を説明する図である。
- 図28は、インタリープ・パターン記述言語で定義された場合の実現を説明する図である。
- 図29は、インタリープ・パターン記述言語で定義された場合の実現を説明する図である。
- 図30は、インタリープ・パターン記述言語で定義された場合の実現を説明する図である。

図31は、インタリープ・パターン記述言語で定義された場合の実現を説明する図である。

図32は、インタリープ・パターンを生成する手順のフローを示す図である。

図33は、生成されたインタリープ・パターンの決定の手順を示すフローチャートである。

図34は、ターボ符号器の構成例を説明するブロック図である。

図35は、移動無線の送受信機の構成例を説明するブロック図である。

図36は、デインターリービングの例を示す図である。

図37は、デインターリービングの例を示す図である。

図38は、ターボ符号に適したインタリープ・パターンの作成を説明するフローチャートである。

図39は、インタリープ・パターンの決定過程の詳細を示す図である。

図40は、インタリープ・パターンの決定過程に使用する、予め定めたインタリープ・パターンのリストを示すテーブルである。

図41は、インタリープ・パターンのマルチステージ・インタ

リービング法による作成過程の詳細を示す図である。

図42は、インタリープ・パターンの作成過程の1段を説明する図である。

図43は、図42の作成過程の具体的な例である。

図44は、インタリープ・パターンの作成過程の他の例を説明する図である。

図45は、作成されたインタリープ・パターンのチェックを説明する図である。

。

図46は、伝送路インタリーバに適したインタリープ・パターンの作成を説明するフローチャートである。

図47は、インタリープ・パターンの決定過程の詳細を示す図である。

図48は、インタリープ・パターンの作成に使用する、予め定めたインタリープ・パターンのリストを示すテーブルである。

図49は、インタリープ・パターンのマルチステージ・インタリービング法による作成過程の詳細を示す図である。

図50は、インタリープを行う装置の例を示す図である。

#### 発明を実施するための最良の形態

以下、図面を参照して本発明の実施の形態を詳細に説明する。以下の説明では、2次元の配列を用いてインタリーバを図示し説明しているが、これは説明の便宜上のものである。2次元の配列ではなく1次元の配列を用いて本発明を実施できることは言うまでもない。

なお、以下のインタリービング処理等は、入力されるビット系列等の信号処理等を行う処理装置において実行される。

まず、本発明の第1の実施の形態を説明する。

図2と図3は、本発明の第1の実施の形態のインタリービング方法を示す。図2は、まず、従来例と同様に1フレームのデータ200として1152ビットを入力して、 $72 \times 16 (= 1152)$  イ

ンタリーバ210のバッファに対して行方向に行ベクトル215の書き込みを行う。本発明のインタリービング方法は、 $72 \times 16$  インタリーバではなく一般的の  $N \times M$  インタリーバについても適用可能であることはもちろんである。 $72 \times 16$  インタリーバ210において、16列ある列ベクトル220等それぞれは72ビットで構成されているが、この列ベクトル220等を読み出し、それぞれについて対応する16個の $9 \times 8 (= 72)$  インタリーバ230、235、…、240でインタリービングを行う。つまり $9 \times 8$  インタリーバ230等のバッファに対して上記の列ベクトル220等を行方向に書き込む。最後に16個ある $9 \times 8$  インタリーバの各々に対して、順に列方向にデータを読み出し、出力すなわちインタリープされたデータ245を取り出す。

図1におけるインタリープされたデータの配置は、列の中に関しては時間順に順序正しく並んでいるが、図2におけるインタリープされたデータの配置は、より複雑なものになっていることが分かる。ここで定量的に評価するため、従来の技術で説明した尺度(2)の観点でインタリービングの能力を評価する。

図1の出力130において、「0」の隣は「16」である。したがって、入力データにおいては、この2つのビットは16ビット離れている。全てのビットに

ついて同様に調べてみると、連続する2ビットの出力は、入力系列において少なくとも16ビット離れていることが分かる。つまり、上記尺度(2)は16ビットとなる。

図2では、上記尺度(2)は128ビットとなり、本実施の形態1の方法は、インタリーピングの能力を引き上げることができたことが分かる。

図3は、さらにもう一段階インタリーピングを繰り返している方法を示す。図2の $9 \times 8$ インタリーバ $230$ 等内の各列のデータをさらに、各列毎に、 $3 \times 3$ (=9)インタリーバ $285, 287, \dots, 293$ において、インタリーピングを行っている。最終的に

$3 \times 3$ インタリーバ $285$ 等に対して、順に列方向にデータを読み出し、出力するわちインタリープされたデータ $295$ を取り出す。

図3では、上記尺度(2)は384ビットなり、繰り返しインタリーピングを行うことでインタリーピングの能力を引き上げることができることが分かる。

次に、本発明の第2の実施の形態を説明する。

図4は、本発明の第2の実施の形態のインタリーピング方法を示す。

図3では、 $72 \times 16$ インタリーバ $310$ を列方向に読み出していたが、本実施の形態2では $72 \times 16$ インタリーバ $310$ の行ベクトル $315$ の16ビットを読み出し、 $4 \times 4$ インタリーバ $320, 330$ 等に同様に行方向に書き込みを行う。次に、 $4 \times 4$ インタリーバ $320$ 等の列を順に読み出し、先ほどの $72 \times 16$ インタリーバ $310$ の各行にデータを戻す。この場合、先の $72 \times 16$ インタリーバ $310$ の代わりに別の $72 \times 16$ インタリーバ $335$ を用いてもよい。すべての $4 \times 4$ インタリーバ $320$ 等の列のデータを戻したら、 $72 \times 16$ インタリーバ $310$ のバッファを列方向に読み出し、インタリープされたデータ $340$ を取り出す。

従来のインタリーピング法の出力結果 $350$ と比較すると、 $72 \times 16$ インタリーバ $310$ における各列内におけるデータ配置は同じだが、各列の配置が異なることが分かる。ここで、上記尺度(1)の観点でインタリーピングの能力を評価する。従来の方法においては、入力データの「0」と「1」は、インダリープ

されたデータにおいては、72ビット離れている。同様に全てのビットについて調べると、上記尺度(1)は72ビットであることが分かる。本実施の形態2では上記尺度(1)は288ビットとなり、インターリービングの能力を引き上げることができたことが分かる。

次に、本発明の第3の実施の形態を説明する。

図5は、本発明の第3の実施の形態のインターリービング方法を示す。

す。本実施の形態は、実施の形態第1と第2とを組み合わせ、それぞれのインターリービング法を繰り返す方法である。

図5において、1フレームのデータ400として16ビットを入力して、 $4 \times 4 (= 16)$ インターバ410のバッファに対して行方向に書き込みを行う。 $4 \times 4$ インターバ410において、4列ある列それぞれは4ビットで構成されているが、この列を読み出し、それぞれについて対応する4個の $2 \times 2 (= 4)$ インターバ420、425、430、435でインターリービングを行う。つまり $2 \times 2$ インターバ420等のバッファに対して上記の列を行方向に書き込む。

次に、 $2 \times 2$ インターバ420等の列の2ビットを列毎に順次読み出し、先ほどの $4 \times 4$ インターバ410の各列にデータを戻す。この場合、先の $4 \times 4$ インターバ410の代わりに別の $4 \times 4$ インターバ440を用いてもよい。

このインターバ410または440に対して各行を読み出し、それぞれについて対応する4個の $2 \times 2 (= 4)$ インターバ445、450、455、460でインターリービングを行う。つまり $2 \times 2$ インターバ445等のバッファに対して上記の行を行方向に書き込む。

次に、 $2 \times 2$ インターバ445等の列の2ビットを列毎に順次読み出し、先ほどの $4 \times 4$ インターバ410または440の各行にデータを戻す。この場合、先の $4 \times 4$ インターバ410または440の代わりに別の $4 \times 4$ インターバ470を用いてもよい。すべての $2 \times 2$ インターバ445等の列のデータを戻したら、 $4 \times 4$ インターバ410または440または470のバッファを列方向に読み出し、インターブされたデータ480を取り出す。

以上により、尺度（1）および（2）を同時に改善することができ、インタリービングの能力をさらに引き上げることができる。

したがって、本発明によれば同じN×Mのバッファ・サイズのイ

ンタリービングであっても、連続する2ビットの入力は、充分繰り返すインタリーブ後の出力系列において $2N$ ビット以上遠くに引き離せ、連続する2ビットの出力は、入力系列においては $2M$ ビット以上離すことが可能である。例えば、 $8 \times 8$ のバッファ・サイズのインタリービングの場合は、3回の繰り返しによりインタリーブ後の出力系列において $2 \times 8$ ビット以上遠くに引き離せ、連続する2ビットの出力は、入力系列においては $2 \times 8$ ビット以上離すことが可能である。

さらに、本発明はバースト誤り伝送路またはバースト誤り記録媒体で生じるバースト誤りのランダム化に用いることができる。ターボ符号化に適用するインタリービング法として用いることもできる。

次に、本発明の第4の実施の形態を説明する。図6は、本実施の形態を示す。

上述の実施の形態第1から第3のように、同一のインタリービング・ステップにおいて必ずしも同一のインタリーバを繰り返し用いなくても良い。図6において、入力系列を $72 \times 16$ インタリーバに書き込んだ後、 $72 \times 16$ インタリーバ600の行毎に16ビットずつ読み出し、第1行目は $4 \times 4$ インタリーバ610に、第2行目は $6 \times 3$ インタリーバ620に、第3行目は $8 \times 2$ インタリーバ630に、等のように行毎にインタリーバの形を変えることも可能である。また、16ビットの入力系列に対して $6 \times 3$ インタリーバというようにインタリーバ・バッファの中のすべてに対して入力データを書き込む必要もない。

次に、本発明の第5の実施の形態を説明する。図7は、本実施の形態を示す。

図7は、図2と同様のインタリービングの例を示す。ただし、図2の例では、第1のインタリービング・ステップにおいて、入力系列を複数の16ビットからなるブロックに分割しているが、そのブロックの中は時間順のままである。一方、図7の例では、そのブ

ロック内において、擬似ランダム・インタリービングを行い、 $72 \times 16$ インタ

リーバ700に書き込んでいる。このように、本発明の繰り返し処理と疑似ランダム・インタリービング方法に用いられるようなビット順の入れ替え処理とを組み合わせることが可能であり、このインタリービング方法の例を示したもののが図7である。

次に、本発明の第6の実施の形態を説明する。

上述した実施の形態においては、ビット単位のインタリービング方法を示したが、シンボル単位でも同様にしてインタリービングを行うことができる。以下、その例を第6の実施の形態として説明する。

まず、 $L$ ビット長のデータを $N \times M$  ( $L \leq N \times M$ ) のブロックインタリーバへ書きこむ。ここで、1シンボルを $K$ ビットとし、 $N \times M$ のブロックインタリーバにおいては連続(隣接)する $K$ ビットを1シンボルとする。

縦方向に連続する $K$ ビットを1シンボルとした第1の場合には、( $N/K$ )シンボル $\times M$ シンボルのブロックインタリーバとみなすことができ、上述した方法で縦方向及び横方向に複数段インタリーブを行い、シンボルをビットに直して読み出すことで、シンボル単位のインタリービングが可能となる。また、横方法に連続する $K$ ビットを1シンボルとした第2の場合には、 $N$ シンボル $\times (M/K)$ シンボルのブロックインタリーバとみなすことができ、同様にしてシンボル単位のインタリービングができる。更に、 $K = N_1 \times M_1$ であるとして、隣接する $K$ ビットを1シンボルとした第3の場合には、

( $N/N_1$ )  $\times$  ( $M/M_1$ ) シンボルのブロックインタリーバとみなして、同様にしてシンボル単位のインタリービングができる。

以上説明した方法の具体例を図を用いて説明する。図8は、 $L = 64$ ビット長のデータが( $N =$ ) 8ビット $\times$  ( $M =$ ) 8ビットのブロックインタリーバに書き込まれたことを示す図である。この後の

処理は、第1～第3の場合について、図9～図11において説明する。

図9は $K = 2$ における第1の場合を示す図である。図9の4シンボル $\times 8$ シンボルのブロックインタリーバがビット単位のインタリービングについて既に説明した方法でインタリービングされ、インタリーバ650のシンボルがビットに直

され、インタリーバ660となり、読み出しが行われる。

また、 $K=2$ における第2の場合は、図10に示す方法でインタリービングが行われ、 $K=4$ における第3の場合は、図11に示す方法でインタリービングが行われる。これらの具体的方法については、それぞれの図及び上記の説明等から明らかであるので説明は省略する。

以上説明したように、本発明の上記第1～第6の実施の形態におけるインタリービング方法によれば、前述の第1の目的が達成され、バッファに対する書き込みまたは読み出しの処理を1回行った後、さらに順番を入れ替える処理を繰り返して適用することにより、1回づつ時間順に読み書きしている場合よりもインタリービングの効果を向上させることが可能である。

なお、このようなインタリービング方法をマルチブル・インタリービング法と称する。

上述したとおり、デジタル・システムの場合、ビットかシンボル等の単位でインタリービングにおける並び替えが行われる。上述の方法は、バッファ等にデータを書き込み、それを読み取る方法を示しているが、インタリービングによる順番の入れ替え情報をパターン（以下、インタリープ・パターンと称する）として持ち、それを参照して並べ替えることも可能である。インタリービングはビット単位、シンボル単位等で処理されるが、以下では簡単のためビット単位に行った例を示す。

図12は、16ビット系列のインタリービングを行った例である。

図12では、インタリープ・パターン・テーブルを参照することによりビット単位のインタリービングを行っている。図12では、インタリービングが行われる入力された16ビットの系列670は、インタリープ・パターン・テーブル680に記憶されている順番にしたがって、入力系列内のビット（もしくはシンボル等）の順番の入れ替えが行われる。

図12において、そこに示されているインタリープ・パターン・テーブルに示されている順番を、矢印のように縦方向の順に0、8、4、12、2、…と読み出す。その読み出された順番に、入力された16ビットの系列を、順次、入

力系列の0番目のビットを出力系列の0番目に、1番目のビットを8番目と入れ替える。そして、インタリービング後のビット系列を出力する。

次に、本発明の第2の目的を達成する手段における実施の形態を、第7の実施の形態から説明する。

図13～図15は、本発明の実施の形態7を示す。

図13および図14では、16ビットの系列670をインタリービングする例を示している。すなわち、4ビット系列の変換を示すインタリープ・パターン・テーブルAおよびインタリープ・パターン・テーブルBを用意しておき、この2つの4ビットのインタリープ・パターン・テーブルから、16(4×4)ビット系列の変換を示すインタリープ・パターン・テーブル680を生成する。そして、この生成された16ビット系列のインタリープ・パターン・テーブルを用いて、入力される16ビット系列670のインタリービングを行う。

図13において、予め用意されているテーブルAがテーブルCの書き込み方向、またはテーブルBがテーブルCの書き込み方向の垂直方向を規定している。LA=16ビットのインタリービングを行う際、LA=4ビットのインタリープ・パターンを記述したインタリープ・パターン・テーブルAとLB=4ビットのインタリープ・

パターンを記述したインタリープ・パターン・テーブルBを用いて、L( $\leq LA \times LB$ )ビットのインタリープ・パターンを記述したインタリープ・パターン・テーブルCを作成する。(図13の①)

インタリープ・パターン・テーブルのC[i] (ビット列Cのi番目)を作成するための演算は、図13の例では、

$$C[i] = A[i \% LA] + LA \times B[i / LA]$$

であり、LA=4である。(ここで、iは系列のビットの位置を示すアドレスであり、0以上の整数とする。これ以後の記述では、系列の位置を示すアドレスをi、j、k、…とし、0以上の整数とする。A[i]は、テーブルAのi番目の要素を示す。”%”は剰余演算子で、i%LAはiをLAで割り算した余りを意味し、また、i/LAはiをLAで割り算した結果の整数部(小数部を切り捨

てた数)を意味する。以下の式においても同様である。但し、テーブルAがテーブルCの横方向のパターンを規定し、テーブルBがテーブルCの縦方向を規定するものとする。

この結果をインタリープ・パターン・テーブルC 680に対して横方向に書き込むことで、インタリープ・パターン・テーブルC 680を生成する。

インタリービングする際は、テーブルCを縦方向に順番に読み出し、それを参照して入力系列のインタリービングを行う(図13の②)。これは、インタリープ・パターン・テーブルC 680に記憶されている順番にしたがって、系列内のビットの順番の入れ替えを行う。

なお、読み出し方向は必ずしも下向きである必要はない。また、書き込み方向は必ずしも右方向である必要が無い。例えば、読み出し方向を上向きというようになくとは逆方向にすることも可能である。

また、図13において、テーブルAがテーブルCの書き込み方向、またはテーブルBがテーブルCの書き込み方向の垂直方向を規定しているとしたが、テーブルAとBの関係は入れ替えることができ、

インタリープ・パターン・テーブルAおよびBは同一のパターンであっても、異なるパターンであっても構わない。

テーブルAとテーブルBが同一の場合、どちらか一方のテーブルのみ用いることも可能である。たとえば、 $A = B$ の場合、

$$C[i] = A[i \% L_A] + L_A \times A[i / L_A]$$

もしくは

$$C[i] = B[i \% L_B] + L_B \times B[i / L_B]$$

と記述することも可能である。これらのこととは、以下、図13から図21まで同様である。

図14は、図13の例と同様の結果が得られる別の演算を示している。図14で行われているインタリープ・パターンC 680の作成は、

$$C[i] = L_A \times B[i \% L_B] + A[i / L_B]$$

であり(図14の①)、図13と異なる。但し、テーブルAがテーブルCの横方

向のパターンを規定し、テーブルBがテーブルCの縦方向を規定するものとする。この結果をテーブルCに対して縦方向に書き込む。

なお、インタリープ・パターン・テーブルAとBは入れ替わってもよい。また、テーブルAおよびBは同一のパターンであっても、異なるパターンであっても構わない。テーブルAとBが同一の場合テーブルはAもしくはBのどちらか一方のみでも構わない。

インタリービングする際は、テーブルCを同（縦）方向に順番に読み出し、それを参照して入力系列のインタリービングを行う（図14の②）。

このように、インタリープ・パターン・テーブルC 680の生成する演算は任意に変化させることが可能である。また、読み出し方向と書き込み方向は必ずしも同一方向である必要が無く、図14に示された方向である必要も無い。テーブルの書き込み方向、読み出し方向も自由に設定することが可能である。

図15は、図13および図14に示したような16ビット・インタリープではなく、15ビット・インタリープを行う例である。

図15において、 $L = 15$ ビットのインタリービングを行う際、 $L_A = 4$ ビットのインタリープ・パターンを記述したインタリープ・パターン・テーブルAと $L_B = 4$ ビットのインタリープ・パターンを記述したインタリープ・パターン・テーブルBを用いて、 $L (\leq L_A \times L_B)$ ビットのインタリープ・パターンを記述したインタリープ・パターン・テーブルC 700を作成する（図15の①）。このインタリープ・パターン・テーブルC 700の作成方法は、図13と同様、

$$C[i] = A[i \% L_A] + L_A \times B[i / L_A]$$

となる ( $L_A = 4$ )。但し、テーブルAがテーブルCの横方向のパターンを規定し、テーブルBがテーブルCの縦方向を規定するものとする。

この結果は、テーブルCに対して横方向に書き込むわけだが、テーブルCは16ビットのインタリープ・パターンを記述しているので、15ビットのインタリービングする際は、テーブルCを縦方向に順番に読み出し、15以上の数を読み込んだ場合はそれを破棄することで、15ビットのインタリービングを行う（図15の②）。

図15のように、テーブルCを読み込むときに15以上の数を読み飛ばすことにより、15ビットのインタリーピングを行うことができるが、テーブルCに対して書き込みをする際に15以上の書き込みを禁止することにより、15ビットのインタリープ・パターンを記述したテーブルCを作成し、これをもとにインタリーピングを行うことも可能である。すなわち、図15において、4ビットのインタリープ・パターンを記憶したテーブルAと4ビットのインタリープ・パターンを記憶したテーブルBを用いて、15( $< 4 \times 4$ )ビットのインタリープ・パターン・テーブルCを作成する場合、14(=15-1)以上の数を書き込まないことにより、15ビッ

トのインタリープ・パターン・テーブルを作成することが可能である。

第7の実施の形態においては、例えば、一つのインタリープ・パターン・テーブル(A=Bの場合)、もしくは二つのインタリープ・パターン・テーブルから、よりインタリーピング長の大きいインタリープ・パターン・テーブルを作成している。これにより、あるインタリープ長のパターンをより小さいインタリープ長のパターンの複数の組み合わせで表わすことができ、固定長パターンのメモリ量を削減できる。

たとえば、1000ビット分のインタリープ・パターン・テーブルは、従来では1000ビット分のインタリープ・パターン・テーブルのためのメモリが必要だが、実施の形態7においては、1000ビット・インタリープ・パターン・テーブルを20ビット・インタリープ・パターン・テーブル×50ビット・インタリープ・パターン・テーブルで表現することで、70(=20+50)ビット分のメモリに削減することができる。

また、900ビット・インタリープ・パターン・テーブルを20ビット・インタリープ・パターン・テーブル×50ビット・インタリープ・パターン・テーブルで表現することで、固定長のインタリープ・パターン・テーブルを増加させずに、1000ビットと900ビットのインタリーピングを行うことができ、そのインタリーピング長に対応したインタリーピングを行うことが可能である。

次に第8の実施の形態を説明する。

本発明の第8の実施の形態は、図13～図15に示した第7の実施の形態と同様のインタリーピング結果と同様の結果が得られるが、インタリープ・パターン・テーブルCは作成せず、直接テーブルAとテーブルBから演算して、インタリープ先を求め、これによりインタリーピングを行うものである。第8の実施の形態を図16および図17に示す。

図16においては、図13の例と同様の結果が得られる16ビット系列の例を示している。

図示されているように、図13のようなインタリープ・パターン・テーブルCは作成していない。4ビットのインタリープ・パターンを記憶したテーブルAと4ビットのインタリープ・パターンを記憶したテーブルBを用いて、系列内のビットの入れ替え先を計算し(図16の①)、その結果をもとに、インタリーピングを行う(図16の②)。

この直接テーブルAとテーブルBから系列内の1番目の要素に対するインタリーピング先j番目を計算するための式は、図14における式と同様の

$$C[i] = 4B[i \% 4] + A[i / 4]$$

である。

なお、テーブルAとBは入れ替わってもよい。また、インタリープ・パターン・テーブルAおよびBは同一のパターンであっても異なるパターンであっても構わない。テーブルAとBが同一の場合、

テーブルはAもしくはBのどちらか一方のみでも構わない。つまり、

$$\begin{aligned} j &= 4A[i \% 4] + A[i / 4] \\ &= 4B[i \% 4] + B[i / 4] \end{aligned}$$

でよい。

図17においては、インタリープを指定するのに、インタリープ・パターン・テーブルA、Bを用意する代わりに、インタリープ・パターン方程式(710、720)を用いている。そして、このインタリープ・パターン方程式を、インタリーピングを行うときに演算する(図17の①)ことにより、インタリープを行う(図17の②)。

4ビットのインタリープ・パターンを記述した方程式a710と方程式b720は、

$$j_a = 2(i_a \% 2) + (i_a / 2) = f_a(i_a)$$

ただし、 $0 \leq i_a \leq 4$

$$\begin{aligned} &= 2(i_b \% 2) + (i_b / 2) \\ &= f_b(i_b) \end{aligned}$$

ただし、 $0 \leq i_b \leq 4$ である。

この4ビットのインタリープ・パターンを記述した方程式aと方程式bとの双方を順次用いることにより、16ビット系列内のi番目の要素に対するインタリービング先j番目を計算する。それは、結局、

$$\begin{aligned} j &= 4f_a(i \% 4) + f_b(i / 4) \\ &= 8((i \% 4) \% 2) + 4((i \% 4) / 2) \\ &\quad + 2((i / 4) \% 2) + ((i / 4) / 2) \end{aligned}$$

を計算して、インタリービングを行うことと同じである。

なお、方程式aとbは入れ替わってもよい。インタリープ・パターン方程式aおよびbは同一の式であっても異なる式であっても構わない。方程式aとbが同一の場合、方程式aもしくはbのどちらか一方のみを用いてもよい。

図17に示されているインタリービングにおいて、用意されている4ビットの式から16ビットのインタリープを計算し、その計算結果をまずテーブルに書き込み、インタリープ・パターン・テーブルを作成してからインタリービングを行うことも可能である。これは、第7の実施の形態において、インタリープをテーブルで規定せず、式により規定したことと同じである。

第8の実施の形態においては、一つのインタリープ・パターン・テーブル(A=Bの場合)、もしくは二つのインタリープ・パターン・テーブルから、大きいインタリービングを行っているが、別のインタリープ・パターン・テーブルを生成する必要がないという特徴がある。

また、一つのインタリープ・パターン方程式(a=bの場合)も

しくは二つのインタリープ・パターン方程式から、よりインタリービング長の大きいインタリービングを行うことができるが、別のインタリープ・パターン・テーブルを生成する必要はない。ただし、インタリープ・パターン・テーブルを生成することも可能である。

次に、第9の実施の形態を説明する。

第9の実施の形態は、第7の実施の形態または第8の実施の形態で示される処理を複数回繰り返して用いている例である。第9の実施の形態を図18～図22を用いて説明する。

図18は、第7の実施の形態である図13で示される処理を複数回繰り返して用いて、2つの2ビットのインタリープ・パターン・テーブルA、Bおよび4ビットのインタリープ・パターン・テーブルCから16ビットのインタリープ・パターン・テーブルを生成する例である。

図18において、 $L = 16$ ビットのインタリービングを行う際、まず、 $L_A = 2$ ビットのインタリープ・パターンを記憶したテーブルAと、 $L_B = 2$ ビットのインタリープ・パターンを記憶したテーブルBを用いて、 $L_C = 4 \leq L_A \times L_B$ ビットのインタリープ・パターン・テーブルD730を生成する（図18の①）。

。

インタリープ・パターン・テーブルD730を生成するための演算は、例えば

$$D[i] = 2A[i \% 2] + B[i / 2]$$

（テーブルAがテーブルDの書き込み方向、またテーブルBがテーブルDの書き込み方向の垂直方向を規定している場合）

で行うことができる。

次に、作成したテーブルDと $L_D = 4$ ビットのインタリープ・パターンを記憶したテーブルCを参照することにより、 $L_E = 16 \leq L_C \times L_D$ ビットのインタリープ・パターン・テーブルE740を生成する（図18の②）。

インタリープ・パターン・テーブルE740を生成するための演

算は、例えば、

$$E[i] = 4D[i \% 4] + C[i / 2]$$

(テーブルDがテーブルEの書き込み方向、またテーブルCがテーブルEの書き込み方向の垂直方向を規定している場合)  
で行うことができる。

このようにして生成したインタリープ・パターン・テーブルEを参照することにより、16ビット系列のインタリーピング処理を行う(図18の③)。

なお、テーブルAとBは入れ替わってもよい。インタリープ・パターン・テーブルA、B、Cは同一のパターンであっても異なるパターンであっても構わない。同一パターンの場合、どちらか一方のテーブルのみ用いてもよい。テーブルAとBが同一の場合、

$$\begin{aligned} C[i] &= A[i \% 4] + 4A[i / 4] \\ &= B[i \% 4] + 4B[i / 4] \end{aligned}$$

となる。また、読み出し方向と書き込み方向は必ずしも同一方向である必要が無い。図に示された方向である必要も無い。

図19は、第8の実施の形態である図16に示した処理を複数回繰り返し組み合わせて、2つの2ビットのインタリープ・パターン・テーブルA、Bおよび4ビットのインタリープ・パターン・テーブルCを用いて、16ビットのインタリーピングを行う例である。

図19において、まず、4ビットの  $i_d$  ( $0 \leq i_d < 4$ ) 番目のビットの入れ替え先  $j_d$  番目を、テーブルAとBを参照して計算する(図15の①)。この演算は、例えば、

$$j_d = 2A[i_d \% 2] + B[i_d / 2]$$

(テーブルAがテーブルCの書き込み方向、またテーブルBがテーブルCの書き込み方向の垂直方向を規定している場合)である。

次に、入力系列16ビット750の  $i$  ( $0 \leq i < 16$ ) 番目のビットの入れ替え先  $j$  番目を、上述の演算結果とインタリープ・パターン・テーブルCを参照して計算する(図19の②)。この演算

は、例えば、

$$j = 4 \times j_d + C [i/4]$$

(ただし、 $j_d$  は  $i_d = i \% 4$  の移動先)

で行う。

最終的にこの計算結果に基づき、16ビット系列内のビットの順番を入れ替えることで、インタリービングを行う（図19の③）。

なお、テーブルAとBは入れ替わってもよい。インタリープ・パターン・テーブルA、B、Cは、同一のパターンであっても異なるパターンであっても構わない。同一パターンの場合、どちらか一方のテーブルのみ用いてもよい。テーブルAとBが同一の場合、

$$\begin{aligned} C[i] &= A[i \% 4] + 4A[i/4] \\ &= B[i \% 4] + B[i/4] \end{aligned}$$

である。

また、計算結果をテーブルに記憶することで、インタリープ・パターン・テーブルも作成することができる。

図20は、方程式でインタリープを規定している第8の実施の形態の図17の処理を繰り返した例である。

図20において、まず、2ビットのインタリープ・パターンを記述した方程式a ( $j_a = i_a$ ) と方程式b ( $j_b = i_b$ ) から4ビットのインタリープ・パターンを記述するための方程式d、すなわち

$$j_d = 2(i_d \% 2) + (i_d / 2)$$

を作成する（図20の①）。

次に、方程式dと4ビットのインタリービング方程式eから、16ビットのインタリービング方程式e、すなわち

$$\begin{aligned} j_e &= 8((i_e \% 4) \% 2) + 4((i_e \% 4) / 2) \\ &\quad + 2((i_e / 4) \% 2) + ((i_e / 4) / 2) \end{aligned}$$

により計算する（図20の②）。この計算した方程式eを用いてインタリービングを行う（図20の③）。

このとき、計算結果をまずインタリープ・パターン・テーブルに書き込み、テ

ーブルを作成してからインタリーピングを行うことも可能である。

図21は、方程式とテーブルでインタリーブを規定して、それを基にインタリーブ・パターン・テーブルを作成して、インタリーピングを行う例を示している。

図21において、2ビットのインタリーブ・パターンを記述した方程式aと2ビットのインタリーブ・パターン・テーブルBから4ビットのインタリーピング方程式d、すなわち、

$$j \ d = 2 \ (i \ d \% 2) + (i \ d / 2)$$

を作成する（図21の①）。

次に、方程式dと4ビットのインタリーブ・パターン・テーブルCから、

$$E [i] = 4 \ (2 \ (i \% 4) \% 2) + ((i \% 4) / 2) \\ + B [i / 4]$$

により、16ビットのインタリーブ・パターン・テーブルE760を作成する（図21の②）。そして、テーブルE760を参照してインタリーピングを行う（図21の③）。

図22は、図13の変形であるが、複数の4ビットのテーブルから16ビットのテーブルを作成する例を示している。

図22において、16ビットのインタリーブ・パターン・テーブルC770を作成する場合、複数の4ビットのテーブルA0からA3と4ビットのテーブルBとを用いて、

$$C [i] = A [i / 4] [i \% 4] + 4 B [i / 4]$$

（テーブルA0～A3がテーブルCの書き込み方向、またテーブルBがテーブルCの書き込み方向の垂直方向を規定している場合）によりテーブルC770を作成し（図22の①）、テーブルCを参照することによりインタリーピングを行う（図22の②）。

なお、テーブルA0～A3とBは入れ替わってもよい。インタ

リーブ・パターン・テーブルA0～A3およびBは同一のパターンであっても異なるパターンであっても構わない。読み出し方向は必ずしも下向きである必要は

ない。また、書き込み方向も必ずしも右方向である必要が無い。

次に、第10の実施の形態について説明する。

第10の実施の形態は、インタリープ・パターン記述言語によって定義されたインタリープが与えられた場合、その言語を認識して、上述の第7～9の実施の形態等のインタリープを用いることにより、インタリープ・パターンを作成したり、インタリービングを行うものである。

図23～図26は、インタリープ・パターン記述言語の定義について説明する図である。図27～図31は、図23～図26で定義されたインタリープ・パターン記述言語によってかかれた式が与えられた場合、その言語を認識し、上述の第7～9の実施の形態いずれかまたは組み合わせ等を用いて、インタリープ・パターンを作成し、インタリービングを行う例である。図32はインタリープ・パターンの自動作成を行うことを説明する図である。図33は、インタリープ・パターンの決定を行うフローを説明するフローチャートである。

まず、図23～図26を用いて、インタリープ・パターン記述言語について説明する。

図23は、インタリープ・パターン記述言語の定義1：L [N×M] を説明している。L [N×M] と記載されている場合は、N×Mブロック・インタリーバを意味する。このインタリーバは、Lビットの系列をN×Mのブロック・インタリーバでインタリービングを行うことを意味している。例として、L [N×M] のブロック・インタリーバにより、Lビットの系列がインタリープされている様子が示されている。

図24は、定義2：R {A} を説明している。R {A} と記載されている場合は、Aビットを逆順に並び替えることを意味している。例として、R {6} により、6ビットの系列が逆順に並び替えられていることが示されている。

図25は、定義3：L [N<sub>1</sub>×M<sub>1</sub>、N<sub>2</sub>×M<sub>2</sub>、…] を説明している。L [N<sub>1</sub>×M<sub>1</sub>、N<sub>2</sub>×M<sub>2</sub>、…] と記述されているときは、複数の系列（各系列はLビット）をそれぞれ対応するインタリーバでインタリービングを行うこと

を意味する。例として、 $6 [3 \times 2, 2 \times 3]$  と記述されている場合、2つの6ビットの系列がそれぞれインタリーピングされている様子が示されている。

図26は、定義4： $L [N_1 [N_2 \times M_2] \times M_1]$  を説明している。

$L [N_1 [N_2 \times M_2] \times M_1]$  と記述されると、Lビットの系列を  $N_1 \times M_1$  ブロック・インタリーバでインタリーピングを行った後、 $M_1$  個ある縦配列 ( $N_1$  ビット) のそれぞれを  $N_2 \times M_2$  インタリーバでインタリーピングを行うことを意味する。

また、 $L [N_1 \times M_1 [N_2 \times M_2]]$  と記述されている場合は、Lビットを  $N_1 \times M_1$  ブロック・インタリーバでインタリーピングを行った後、 $N_1$  個ある横配列 ( $M_1$  ビット) のそれぞれを  $N_2 \times M_2$  ブロック・インタリーバでインタリーピングを行うことを意味する。

例として、 $16 [4 [2 \times 2] \times 4]$  と記述されている場合が示されている。このように記述されている場合は、図示のように、16ビットの系列780が、 $4 \times 4$  のブロック・インタリーバA790に書き込まれ、その縦配列が読み出されて、それぞれ4つの $2 \times 2$  のインタリーバB～Eにより、インターリピングされていることを意味している。インタリーバF800は、インタリーバB～Eの結果をまとめるために用いられているものである。

上述において説明したインタリーピング記述言語は、インターリープ・パターンを生成するために、また、インターリープ・パターンを

参照して入力系列のインタリーピングを行うために、用いることができる。

図27～図31を用いて、上述したインターリープ・パターン記述言語で表されたものがどの様に実現されているのかを説明する。

図27は、インターリープ・パターン記述言語により、 $16 [4 [2 \times 2] \times 4 [2 \times 2]]$  と表されたインターリープ・パターンの作成要求を、前述したインターリープ方法により、実現した例を示している。

$16 [4 [2 \times 2] \times 4 [2 \times 2]]$  で記述されているインターリープ・パターンの意味は、以下の通りである。

(a) 1段目のインタリーバは $4 \times 4$  ブロックインタリーバで16ビットのイ

ンタリービングを行う。

(b) 1段目のインタリーバの各横配列(4ビット)は $2 \times 2$ インタリーバでインタリービングされる。

(c) 1段目のインタリーバの各縦配列(4ビット)は $2 \times 2$ インタリーバでインタリービングされる。

これを、どの様にして実現しているかを図2.7は説明している。

図2.7において、入力された16ビット系列810は、 $4 \times 4$ のブロック・インターバA820に書き込まれる(図2.7の①)。次に、インターバA820から横方向に読み出されて、それぞれ $2 \times 2$ のインターバB～Eによりインタリービングされる(図2.7の②)。これを、インターバF830に書き込み(図2.7の③)、今度は縦方向に読み出して、それぞれ $2 \times 2$ のインターバG～Jによりインタリービングされる(図2.7の④)。

その結果をテーブル840に書き込む(図2.7の⑤)ことで、記述されたインターブ・パターンを作成することができる。

図2.4は、インターブ・パターン記述言語により、図2.3のときと同様の16 [ $4 [2 \times 2] \times 4 [2 \times 2]$ ]と表されたインターブ処理要求を、前述したインターブ方法により、実現した例

を示している。

図2.8において、図2.8の①から⑤の処理により記述されたインターブ・パターンを作成するまでは、図2.7の①から⑤の処理と同様であるので説明は省略する。この後に、作成されたインターブ・パターン・テーブル850を参照することにより、インターリビングを行う(図2.8の⑥)ことで、記述されたインターリビングを行うことができる。

図2.9および図3.0は、上述したインターブ・パターン記述言語によってかけられた式が与えられた場合、その言語を認識し、第7～9の実施の形態またはそれらの組み合わせ等を用いて、インターブ・パターンを作成し、インターリビングを行った例である。

図2.9は、16 [ $4 [2 \times 2] \times 4 [2 \times 2]$ ]と記述されたインターブ・

パターンの作成要求（もしくはインタリープ処理要求）を、例えば、上述の13で説明したインタリービング処理を繰り返して用いて行うことを示している。

図29では、インタリープ・パターン記述言語で表されたインタリープ・パターンを実現するために、まず、インタリープ・パターン・テーブルA～Dを用いて、インタリープ・パターン・テーブルEおよびFを作成する（図25の①および②）。次に、インタリープ・パターン・テーブルEおよびFを用いて演算することにより、インタリープ・パターン・テーブルG860を作成する（図29の③）。要求がインタリービングを行うことである場合は、このインタリープ・パターン・テーブルGを用いてインタリービングを行う（図29の④）。

図30は、同じく $16 [4 [2 \times 2] \times 4 [2 \times 2]]$ と記述されたインタリープ・パターンの作成要求（もしくはインタリープ処理要求）を、例えば、上述の図17で説明した処理を繰り返すことで実現していることを示している。

図30において、2ビットのインタリープ・パターン方程式a～

dにより、4ビットのインタリープ・パターン方程式e、fを作成する（図30の①、②）。次に4ビットのインタリープ・パターン方程式e、fから16ビットのインタリープ・パターン方程式gを作成し（図30の③、④）、これを用いて、インタリービングを行う（図30の⑤）。

生成されたインタリープ・パターンは、一度記憶すれば、次回もしくは別のインタリーバにおいて、もう一度作り直す必要が無く、インタリープ・テーブルを読み込むだけで同じインタリービング処理を行うことができる。

これを説明するのが図31に示した例である。

図31では、同じく $16 [4 [2 \times 2] \times 4 [2 \times 2]]$ と記述されたインタリープ処理要求を行う場合で、例えば、すでに $4 [2 \times 2]$ という4ビットのインタリープ・パターンがすでに保持されていることが前提である。

例えば、図31においては、図29と同様の実現方法をとるとする。このとき、すでに $4 [2 \times 2]$ というインタリープ・パターンをシステムが、すでにインタリープ・パターン・テーブルAおよびBという形態で保持している。このため、 $4 [2 \times 2]$ に対応する処理（図29の①および②）を行わずに、保持してい

るインタリープ・パターンを参照してインタリープ・パターンを作成し(図31の①)、作成したインタリープ・パターン・テーブルCを参照してインタリーピングを行う(図31の②)。

このようにして、保持されている4[2×2]のインタリーバから16[4[2×2]×4[2×2]]を生成することができる。

図27～図31で示したインタリープ・パターン記述言語で表されたインタリープ・パターンの実現方法は、これに限るものではなく、例えば、これらの処理は組み合わせて行うことも可能である。従って、インタリープ・パターンは、インタリープ・パターン・テーブルやインタリープ・パターン方程式等はどちらを用いててもよ

い。

さて、上述のインタリープ・パターンの生成方法について図32に示すフローチャートを用いて説明する。

図32において、インタリーピング長レビットが与えられたとき、まず一段目( $L \leq N_1 \times M_1$ )のインタリープ・パターンを決定する(S102)。次に、一段目のインタリーバの縦および横にそれぞれ対応する複数の二段目のインタリーバのインタリープ・パターンを決定する(S104)。それぞれの二段目のインタリーバに対応する三段目のインタリーバのインタリープ・パターンを同様に決定する(S106)。この処理を任意の段もしくはインタリーピングができないくなるまで繰り返し(S108)、インタリープ・パターン(インタリープ・パターン記述言語で記述されていてもよい)を生成する(S110)。

各段のインタリーバのインタリープ・パターンの決定法は、因数分解による方法、リストを参照して決定する方法、各段のインタリーピング長の大きさを平方してそれに近い実数に決定する方法、また、それぞれの方法において各段の $N \times M$ インタリーバのNまたはMの値に奇数または素数を選ぶ方法などを用いることができる。

なお、この方法をマルチステージ・インタリーピング法と称する。

上述により生成したインタリープ・パターンから使用に適したものを見別する

フローチャートを示したのが図33である。

図33において、図32で説明したように、インタリービング長に対応するインタリープ・パターンを生成し(S204)、生成されたインタリープ・パターンを検査する(S206)。

検査に不合格であれば、図32に示した各段のインタリーバのパターンの一部もしくは全てを変更し、インタリープ・パターンを再生成して、新たなインタリープ・パターンを生成する(S204)。

検査に合格するまでこれを繰り返し、最終的に生成するインタリープ・パターンを決定する。

検査の項目としては、バースト誤り耐性の強さ、インタリープされたビットのランダム性の強さなどである。特にターボ符号インタリーバとして使用することを前提とする場合は、符号重みの検査、トレリス終端を前提とした符号重みの検査などが挙げられる。

上述の説明においては、シンボルやユニット単位等、インタリービングされるどのような単位にも適用されることが当然である。また、インタリービング対象の系列の長さは時間ごとに変化する場合もある。

第7～10の実施の形態で説明した本発明のインタリープ・パターン生成法によれば、メモリ量の使用を抑えることができるとともに、インタリープ・パターンを保持していないインタリービング長に対しても、柔軟に対応することできる。すなわち、この方法を使用しなければ、1000ビットのインタリービングを行う場合、1000ビットそれぞれのビットの入れ替え方を記述したテーブルが必要であり、インタリープ長（順番の入れ替え対象となるビット、シンボル等の単位の総数）が大きくなると、インタリープ・パターン・テーブルを格納するためのメモリ量が増大してしまう。また、インタリービング長が変化した場合、変化しただけの種類のインタリープ長に対応する複数のインタリープ・パターンをあらかじめ用意している必要がある。そのため、インタリープ長の種類が多くなると、それぞれのインタリープ長のインタリープ・パターン・テーブルを格納するメモリ量が増大してしまう。例えば、インタリービング長が10ビット、10

0ビット、1000ビット、10000ビットと変化するインタリーバにおいては、インタリーブ・パターンを記憶するのに合計 $10 ([\log (10-1)] + 1) + 100 ([\log (100-1)] + 1) + 1000 ([\log (1000-1)] + 1) + 10000 ([\log (10000-1)] + 1)$ ビット分のメモリが必要となる（ただし、 $[\log X]$ は、Xの2の対数をとり、小数部分を切り捨てる意味を意味し、

$[\log (X-1)] + 1$ により、整数Xを2進数で表したときの桁数を表している）。本発明では、このような問題点が発生しない。

また、インタリーブ・パターン記述言語を用いてインタリーブ・パターンを記述して、この様に記述されたインタリーブ・パターンを生成することができる。

この生成されたインタリーブ・パターンは、その特性を調べることも可能で、特性が悪いと判断されたインタリーブ・パターンは特性のよいインタリーブ・パターンに自動的に再生成されるシステムを付加することができる。

次に、本発明における第3の目的を達成するための手段についての実施の形態を説明する。以下では、ターボ符号化器、移動通信等における送受信機の伝送系等に適したインタリーブ方法について説明する。実施の形態を説明するにあたり、ターボ符号化器及び移動通信等における送受信機の構成について説明する。

図34は、ターボ符号化器の構成例を示す図である。ターボ符号化器は、再帰的組織畳み込み符号化器（RSC）（図34（b）参照）を用いて構成されている。図34（a）に示されている例のように、ターボ符号化器入力dに対して、出力X1～X3を出力しているが、冗長ビットX1とX2との相関性を少なくするために、再帰的組織畳み込み符号化器（RSC）13の前にインタリーバ11を挿入している。また、図には示されていないが、ターボ復号器は2つのデコーダ、インタリーバ、インタリーバの逆の処理を行うデインタリーバから構成されている。

図35は、移動通信におけるCDMA方式の送受信機等の構成の一部を示す図である。送信側では、チャネル・エンコーダ21でチャネル符号化を行った後、チャネル・インタリーバ22においてインタリービングを行い、SS送信機23

で、変調した信号にパイロット・シンボルを時分割多重して、拡散変調を行う。

受信側では、R A K E 受信機 2 5において、逆拡散を行った後に、パイロット・

シンボルを用いたR A K E 合成を行い、チャネル・デインタリーバ 2 6においてデインタリービングを行い、チャネル・デコーダ 2 7において復号を行う。伝送系に上記ターボ符号を適用する場合に、チャネル・エンコーダ 2 1にターボ符号化器、チャネル・デコーダ 2 7にターボ復号化器が使用される。

これらの装置において用いられているインタリーバのテーブルは、例えば、図 1 2において説明したもののが使用できる。

ここで、チャネル・デインタリーバ 2 6等で処理が行われているデインタリービングの例について図 3 6 および図 3 7 を用いて説明する。

図 3 6 の (a) はテーブルA ( $L_A = 3$  ビット) 及びテーブルBを用いて 1 2 ビットの系列をインタリービングする例を示している。インタリープ・パターン・テーブルのC [ i ] (ビット列Cの i 番目) を作成するための演算は、

$$C [ i ] = LB \times A [ i \% L_A ] + B [ i / L_A ]$$

であり、 $L_A = 3$ 、 $LB = 4$ である。また、テーブルAがテーブルCの縦方向のパターンを規定し、テーブルBがテーブルCの横方向を規定する。この演算によりインタリープ・パターン・テーブルCを生成し、入力に対してインタリープ・パターン・テーブルCを参照して出力が得られる。

図 3 6 の (b) が、上記の逆の処理であるデインタリープを示す図である。デインタリープのためのデインタリープ・パターン・テーブルC 8 7 0の生成は、上記のテーブルAとテーブルBを入れ替えて、同様の演算を行うことにより行われる。ここで、入力が、上記の 0、8、4、2、…、7 あるとすると、出力は上記の入力である 0、1、2、…、1 1となる。

図 3 7 の (a) は、 $L_A \times LB > L$  の場合の例であり、図 3 7 の (b) はそのデインタリープを示す図である。テーブル生成までの処理は図 3 6 と同様であるが、図 3 7 の (a)においては L 以上の

値を読み出さないか、L 以上の値をテーブル生成時に書き込まない。

図37の(b)における演算式は次のようになる、

$$C[i] = LB \times A[i \% LA] + B[i / LA] - \alpha \quad /$$

但し、 $\alpha$ は以下の規則に従う(C言語による表記である)。

```

 $\alpha = 0$ 
for(j=0, J<(LA × LB-L), J++) {
    if  C[i]>=LB×A[(L-1-j)%LA]
        +B[(L-1-j)/LA]
     $\alpha++;$ 
}

```

以上説明したデインタリープの方法は、前述したインタリープに関して適用でき、以下で説明するインタリープに関しても適用できる。

次に、第11の実施の形態について説明する。

なお、以下、インタリープ・パターンを記述するための表記は、図23～図25において説明したものを使用する。

以下、図38～図45を用いて、ターボ符号に適するインタリープ・パターンの生成方法について説明する。

図38は、Lビットのインタリープ長を有するターボ符号に適したインタリープ・パターンの作成方法を説明するフローチャートである。

図38における第1段階(S302)から高次段階(S306)において、図39で詳細に説明しているような決定過程により、Lビットのインタリープ長を有するターボ符号に対するインタリープ・パターンを決定している。各決定過程のブランチにおける処理がすべて終了することによりインタリープ・パターンが決定されると、決定された最終結果により、インタリープ・パターンを作成する(S308)。そして、作成したインタリープ・パターンをチェックして、Lビットのインタリープ長を有するターボ符号に適したイ

ンタリープ・パターンを得ることができる。

さて、まずインタリープ・パターンの決定過程を説明する。図39はインタリープ・パターンの決定過程の詳細を示す図である。図40は、インタリープ・パ

ターンの決定過程に使用する、予め定めたインタリープ・パターン (P I P) のリストを示すテーブルである。図40に示されている予め定めたインタリープ・パターン・リストは、ターボ符号化に適していることが分かっているインタリープ・パターンのリストである。

図38の第1段階 (S302)において、 $L \leq N^1 \times M^1$  (上付き数字は段階を示す)の行と列として表されるインタリープ・パターンを決定する必要があるが、この第11の実施の形態においては、 $N^1$ は7に固定している。そのため、 $M^1$ は、 $L$ を7で割り、その値が整数となる場合はその値、それ以外の場合はその値より大きい最小の整数とする。なお、 $N^1$ に対する予め定めたインタリープ・パターン (P I P) として、図40のT7として示されているように、R | 7 [3 × 3 [2 × 2] ] } と定めておく。

第1段階におけるインタリープ・パターンの行と列を決定するためには、7つの行 ( $M_1^1 \sim M_7^1$ ) をそれぞれ行と列として表す必要がある。このため、第2段階において、それぞれの行に対応した7つのブランチにおいて、また行と列を決定する必要がある (図39参照)。この行と列を決定するためには、図38の第2段階 (S304) に示されているように、列の数  $M_Y^2$  ( $Y = 1, 2, \dots, 7$ ) はそれぞれ、図39のP I Pのリスト中に示されている、7、13、17、29、37、43、59 ( $L > 3000$  ビット) か、5、7、11、13、17、37、43 ( $3000 > L \geq 301$  ビット) の数から選択する。行の数  $N_Y^2$  は、選択された  $M_Y^2$  で  $M^1$  を割り、得られた値が整数ならその値、それ以外の場合はその値より大きい最小の整数である。この  $N_Y^2$  に対応する I P が図40のテーブル中に定義されている場合は、このブランチに対する演算

は終了する。定義されていない場合は、次の段階 (S306) へ移行する。

次の処理 (S306) では、定義されていない行に対して、また行と列を決定する処理が決定されるまで行われる。この処理 (S306) では、図39に示すように、各ブランチにおいて次のように行われる。列の数  $M_Y^2$  ( $Z \geq 3$ ) は、前段の定義されていなかった行の数  $N_Y^{(Z-1)}$  の平方根の値以下の最も大きい図40のP I P (4と6以外) にある数とする。4と6を除外するのは、ターボ符号に

は列の数が奇数、もしくは、大きな値としたほうがよいことが経験的に知られているからである。行の数  $N_Y^Z$  は、前段の行の数  $N_Y^{(Z-1)}$  を上述で定めた  $M_Y^Z$  で割り、得られた値が整数ならその値、それ以外の場合はその値より大きい最小の整数である。この行の数  $N_Y^Z$  に対して、図 40 の PIP が定義されている場合、このブランチの処理が終了する。この処理を各段階の各ブランチがすべて終了するまで行われる。

なお、図 40 に示したインタリープ・パターン以外に、処理を早く終了するために、他の数に対応するインタリープ・パターンを定義しておくこともできる。ただし、この他の数に対応するインタリープ・パターンは、図 38 に示した高次段階による作成法と同様の方法により作成されたものである。図 40 に定義されているインタリープ・パターンが多くなるとそれだけ処理を早く終了することができる。このように図 40 に示したインタリープ・パターンを増加させても、 $M_Y^Z$  ( $Z \geq 3$ ) は、図 40 に示した 2、3、5、7、8、9、11、13、17、20、29、37、43、47、53、59、61 から選択される。

行と列のインタリープ・パターンがすべて定義されると、この定義された行と列によるインタリーバから、インタリープ・パターンを作成する (S3,08)。この処理を図 41 ないし図 44 を用いて詳しく説明する。

図 41 は、インタリープ・パターンの前述したマルチステージ・インタリーピング法による作成過程の詳細を示す図である。この図からも理解できるように、図 39 における各ブランチの処理とは逆に、下位の段階で決定された行および列に対応するインタリープ・パターン (IP) から、上位の行または列に対応するインタリープ・パターン (IP) をそれぞれ決定して、最終的に 1 ビットのインタリープ・パターン (IP) を作成することができる。

行および列のインタリープ・パターン (IP) から、インタリープ・パターン (IP) をどのように求めるのかを図 42 ないし図 44 を用いて詳しく説明する。図 42 は、インタリープ・パターンの作成過程途中の 1 段を説明する図である。図 43 は、図 42 の作成過程の具体的な例である。図 44 は、インタリープ・パターンの作成過程の最終段階を説明する図である。

図42では、下位の処理で決定された列に対する $N_y^{(z+1)}$ ビットのインタリープ・パターン(IP)および行に対する $M_y^{(z+1)}$ ビットのインタリープ・パターン(IP)を用いて、各ブランチごとの $N_y^2 (= N_y^{(z+1)} \times M_y^{(z+1)})$ ビットのインタリープ・パターン(IP)をどのように作成しているかを説明している。

図42において、列に対する $N_y^{(z+1)}$ ビットのインタリープ・パターン(IP)および行に対する $M_y^{(z+1)}$ ビットのインタリープ・パターン(IP)から演算により、各ブランチごとに、 $N_y^2$ ビットのパターン $C' [i]$ を作成している。 $C' [i]$ は $C'$ のパターンの*i*番目の要素を表している。この演算は、

$$C' [i] = M_y^{(z+1)} A [i \% N_y^{(z+1)}] + B [i / N_y^{(z+1)}]$$

である。なお、Aは、列の $N_y^{(z+1)}$ に対応するインタリープ・パターン(IP)、行のBは $M_y^{(z+1)}$ に対応するインタリープ・パターン(IP)であり、“%”は剰余演算子で割り算した余りを取り、“/”は割り算した結果の整数部を取る(少數部を切り捨て)

ことを意味する。

この演算で求めたパターン $C'$ を縦 $N_y^{(z+1)} \times$ 横 $M_y^{(z+1)}$ の容量を有するメモリに対して縦に書き込み、それを縦に読み出して $N_y^2$ ビットのインタリープ・パターンCを得ることができる。そのとき、 $M_y^2 < N_y^{(z+1)} \times M_y^{(z+1)}$ の場合は $N_y^2$ と等しいか大きい数をメモリには書き込まずにおくことで、 $N_y^{(z+1)} \times M_y^{(z+1)}$ より少ない場合もインタリープ・パターンCを得ることができる。これは、書き込みするときには全部書き込み、読み出すときに $N_y^2$ 以上の値を読み飛すことによっても同様にインタリープ・パターンを得ることができる。

この処理について具体的な例を図43に示す。図43は、4ビット・インタリープ・パターン(IP)Aと4ビット・インタリープ・パターン(IP)Bから、 $4 \times 4 = 16$ より少ない15ビットのインタリープ・パターン(IP)Cを得る例を示している。

図43において、4ビット・インタリープ・パターンA = {0, 2, 1, 3} および4ビット・インタリープ・パターンB = {0, 2, 1, 3} から演算によ

り、16ビットのインタリープ・パターン  $C'$  を得て、 $4 \times 4$  のメモリ中に縦に順次書込む。この演算は、

$$C' [i] = 4A [i \% 4] + B [i / 4]$$

である。このとき、演算により得られた値15は、メモリ中には書き込まない。書き込まれたメモリから、同じ方向に順次読み出すことで、15ビットのインタリープ・パターン  $C$  を得ることができる。

このようにして、図41の最終段階から第2段階までの各行と列に対するインタリープ・パターン (IP) を求め、同様に第1段階の第1行から第7行に対するインタリープ・パターン (IP) を求めることができる。そして、最終的にこの第1行から第7行のインタリープ・パターン (IP) からLビットのインタリープ・パターンを求める。

図44は、この第1段階で得られた複数の行からLビットのイン

タリープ・パターンを作成することを説明するための図である。図44では、列に対応するNビットのインタリープ・パターン (IP) とN個の各行に対応するインタリープ・パターン (IP) からLビットのパターン  $C$  を作成している。

図44において、演算は、列であるNをAとし、各行であるM。ないし  $M_{(n-1)}$  をそれぞれB。ないし  $B_{(n-1)}$  とすると、

$$C [i] = MA [i \% N] + B_{(n-1)} [i / N]$$

と、図43における場合とは異なり、Bも変化している。これによりメモリに縦に順次書き込み、縦に順次読み出すことで、Lビットのインタリープ・パターン  $C$  を作成することができる。L < M × N であるときに処理は、図43における説明と同様である。

上述のようにして、Lビット長のインタリープ・パターンを作成することができる（図38のS308）。図38において、次に、作成したインタリープ・パターンをチェックする（S310）。そして、チェックした結果、得られたインタリープ・パターンがリジェクトされると、N=7（表記は、R（7 [3×R | 3]））として、再度インタリープ・パターンを作成し、前のインタリープ・パターンと比較して、よい方を選択する。

図45を用いて、作成されたインタリープ・パターンのチェックを説明する。

図45において、インタリーピングする前のLビットのビット列と、インタリーピングした後のLビットのビット列とを比較して、インタリーピングする前のビット列の最終ビットから30ビット以内のビットの一部もしくはすべてが、インタリーピングした後の最終のビットから30ビット以内にインタリーピングされると、このインタリーピングを行うインタリープ・パターンはリジェクトされる。

この図38のフローチャートの処理でえられたインタリープ・パターンは、ターボ符号化に適したものである。本ターボ符号化は拘束長3のターボ符号を想定している。つまり図34で示されるよう

に、遅延素子Dが(16)、(17)のように2つ(拘束長-1)あることを示している。拘束長4のターボ符号を想定した場合、図38の第1段階の $N^1$ は8(PIP表記はR 18 [4 [2×2] × 2] |)とすることもできる。つまり $M^1 = L/8$ となり、図41の $M^1$ のブランチは7から8に増える。

次に、本発明第12の実施の形態について説明する。

図46～図49を用いて、本発明の第12の実施の形態である伝送路インタリーバに適したインタリープ・パターンの生成方法について説明する。

図46は、伝送系に適したインタリープ・パターンの作成を説明するフローチャートである。図46における第1段階(S402)から高次段階(S406)において、図47で詳細に説明しているような決定過程により、Lビットのインタリープ長を有する伝送路インタリーバに適したインタリープ・パターンを決定している。各決定過程のブランチにおける処理がすべて終了することによりインタリープ・パターンが決定されると、決定された最終結果により、インタリープ・パターンを作成する(S408)。

さて、まずインタリープ・パターンの決定過程を説明する。図47はインタリープ・パターンの決定過程の詳細を示す図である。図48は、インタリープ・パターンの決定過程に使用する、予め定めたインタリープ・パターン(PIP)のリストを示すテーブルである。図48に定めたインタリープ・パターンは、伝送系に適していることが分かっているインタリープ・パターンである。

図46の第1段階(S402)において、 $L = N^1 \times M^1$ （上付き数字は段階を示す）の行と列として表されるインタリープ・パターンを決定する必要がある。第1段階における列の数である $M^1$ は、1フレームのスロット数（インタリープ長）にしたがって、16、32、64、128の中から選択する。第1段階の行の数 $N^1$ は、選択した $M^1$ で $L$ を割った結果の値以上の最小の整数とする。

第2段階(S404)においては、 $M^2$ は、図48に示されている予め定めたインタリープ・パターン(PIP)のリストに対応する数（“13”および“17”を除く）の中から、 $N^1$ を割りきることができ、かつ $N^1$ の平方根以下の値を持つ最大の整数を選択する。ただし、この整数が $\sqrt{N^1} / 4$  ( $N^1 / 4$ の平方根)より小さかったり、割りきれる候補がPIPリスト中になければ、 $N^1$ の平方根以下の値を持つ最大の整数をPIPリスト（“13”および“17”を除く）の中から選択する。“13”および“17”を除く理由は、経験的に伝送に適したインタリープ・パターンは偶数であるからである。 $N^2$ は、選択された $M^2$ で $N^1$ を割り、その結果の値以上の最小の整数である。第2段階(S404)において、 $N^2$ が図48のPIPのリストに定義されている場合は、インタリープ・パターンがすべて決定されるので、決定されたインタリープ・パターンを用いてLビットのインタリープ・パターンを作成する処理(S408)を行う。PIPのリスト中にはない場合は、次の段階(S406)へ行く（図47参照）。

段階(S406)においては、 $M^2$ （Zは段階の数を示す）は、第2段階と同様に、図48に示されている予め定めたインタリープ・パターン(PIP)に対応する数（“13”および“17”を除く）の中から、 $N^1$ を割りきることができ、かつ $N^1$ の平方根以下の値を持つ最大の整数を選択する。ただし、この整数が $\sqrt{N^1} / 4$  ( $N^1 / 4$ の平方根)より小さかったり、割りきれる候補がPIPリスト中になれば、 $N^1$ の平方根以下の値を持つ最大の整数をPIPリスト（“13”および“17”を除く）の中から選択する。 $N^2$ は、選択された $M^2$ により前段である $N^{(Z-1)}$ を割り、その結果の値以上の最小の数である。このような処理を、 $N^2$ が図48のPIPに定義されているまで行う（図47参照）。定義されている場合は、インタリープ・パターンがすべて決定されるので、決定された

## インタリープ・パターンを用いてレビットのインタリープ・パ

ターンを作成する処理 (S 408) を行う。

なお、図48に示したインタリープ・パターン以外に、処理を早く終了するために、他の数に対応するインタリープ・パターンを定義しておくこともできる。この他の数に対応するインタリープ・パターンの決定は、図46の高次段階の方法と同様に行われる。図48に定義されているインタリープ・パターンが多くなるとそれだけ処理を早く終了することができる。このように図48に示したインタリープ・パターンを増加させても、 $M^2$  ( $Z \geq 2$ ) は、図48に示した2、3、4、5、6、7、8、9、10、11、16、20、32、64、128から選択される。

レビットのインタリープ・パターンを作成する処理 (S 408) を、図49のインタリープ・パターンの前述したマルチステージ・インタービング法による作成過程の詳細を示す図を用いて説明する。

図49において、最終的に得られた、インタリープ・パターン (IP)  $N^2$  および  $M^2$  を用いて、その前の段階の  $N^{(2-1)}$  を求める。この求め方は、図42および図43で説明したものであるので詳細は省略する。これを順次行うことにより、伝送に適したレビットのインタリープ・パターンを得ることができる。

上述において、インタリープ・パターンの長さをビット長としたが、インタリープの対象のインタービング単位がビットであるか他の単位であるかにより、異なる表記となる。

次に、第13の実施の形態である、上述したインタービング方法を使用したインタリープ装置について説明する。

これまでに説明したインタービング方法は、前記の図34～35に示したような機器等におけるインターバル及びデインターバルに適用されるが、インタービングを行う装置であれば、本発明によるマルチプルインタービング法はこれらに限らず適用可能である。但し、第11及び12の実施の形態に示した本発明は特に図34～35に示したような機器、すなわちターボ符号化器、伝送系機

器に適する。

インタリーバとデインタリーバの構成は同一であるので、インタリーバの構成の一例について説明する。図50はインタリープまたはデインタリープを行う装置の例を示し、図50に示すように、インタリーバは入力バッファ30、出力バッファ32、メモリ34、CPU36から構成される。入力バッファ30には入力系列データが記録され、出力バッファ32にはインタリープされた出力系列データが記録される。なお、デインタリープの場合は、入力バッファ30にはインタリープされているデータが記録され、出力バッファ32にはデインタリープされた結果であるインタリープをされる前のデータが記録される。入力バッファ30及び出力バッファ32はRAMやシフトレジスタ等により実現される。メモリ34には、これまで説明したようなインタリープパターンテーブル、または出力バッファ32のインタリープ先アドレスを直接計算するプログラムのどちらか一方または両方が記録されており、RAMやROM等で実現される。CPU36はバッファへの入出力指示、アドレス計算等を行う。なお、上記の構成はLSI等の集積回路で実現することも可能である。

次に、メモリ34にインタリープパターンテーブルのみが記録されている場合の動作について説明する。

入力バッファ30に入力系列データが入力されると、CPU36はメモリ34中のインタリープパターンテーブルを参照して出力先の出力バッファ32のアドレスを読み出し、入力系列データを出力バッフ32アのそのアドレスに出力する。

アドレスを直接計算する場合には、入力バッファ30内の入力系列データのアドレスからCPU36がプログラムにより出力先のアドレスを計算し、出力バッファ32内そのアドレスに出力する。

次に、本発明の記録媒体について第14の実施の形態として説明する。

上述のフローチャート図38、図46等で説明した伝送系及びターボ符号化に適したインタリープ・パターン作成方法をプログラムとしてコンピュータ上で実行することにより、インタリープ・パターンを自動的に求めることができ、前述

のRAM等におけるパターンデータとして使用することができる。このとき、図40および図48に示されている定義されたインタリープ・パターンは、インタリープ・パターン自体を記憶装置上に格納しておきプログラムから参照することも可能であるし、表記のみを記憶し、その都度インタリープ・パターンを作成して用いてもよい。また、例えば、図39や図41および図47や図49で用いている共通の処理はサブルーティンとしておき、他の処理から呼び出すことで使用することができる。

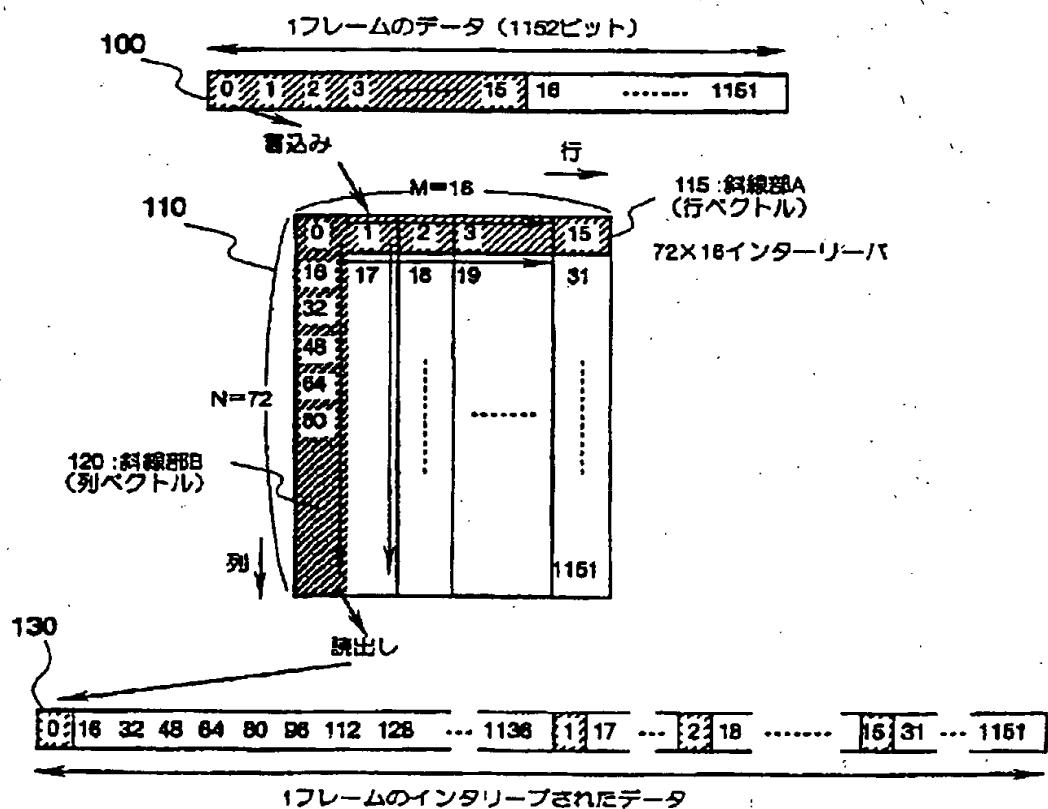
また、本プログラムを格納した記録媒体には、電子メモリ、ハードディスク、光磁気ディスク、フロッピーディスク等があり、これらの記録媒体に記録された本プログラムをコンピュータにローディングする、または記録媒体をコンピュータに組み込むことにより本発明の方法を実施し、インタリープ・パターンを求めることができる。更に、上記の符号／復号化器、送受信器等のメモリに本プログラムをローディングする、または記録媒体を機器に組み込むことにより、自動的に最適なインタリーピングパターンを生成するように符号／復号化器、送受信器等を構成することも可能であり、通信における種々の状況の中で最適なインタリーピング処理を行うことが可能となる。

上記の説明のように、本発明のインタリープ・パターン作成方法を用いることにより、使用目的に適合したインタリープ・パターンを作成することができる。

なお、本発明は、上記の実施例に限定されることなく、請求の範囲内で種々変更・応用が可能である。

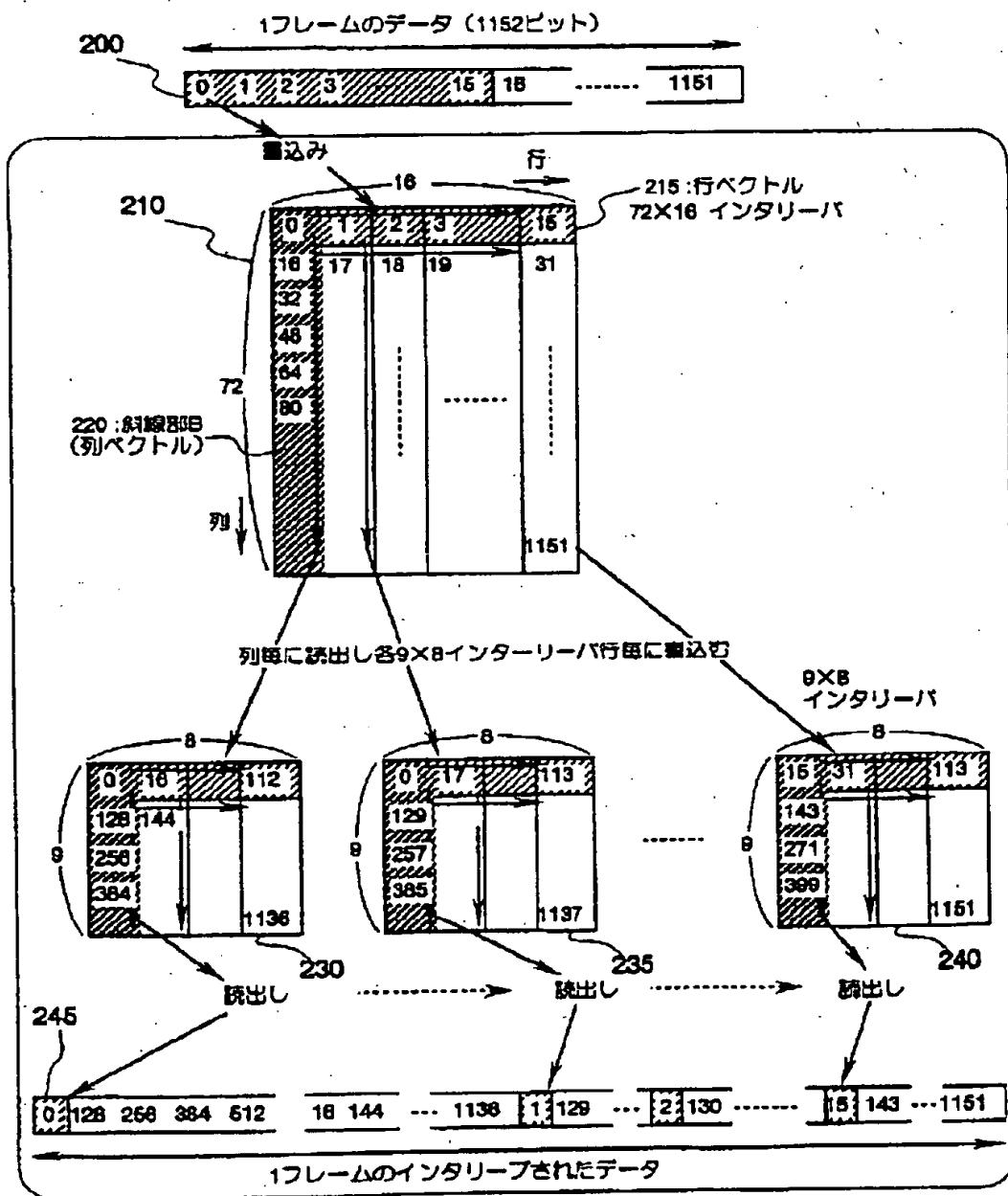
【図1】

FIG.1



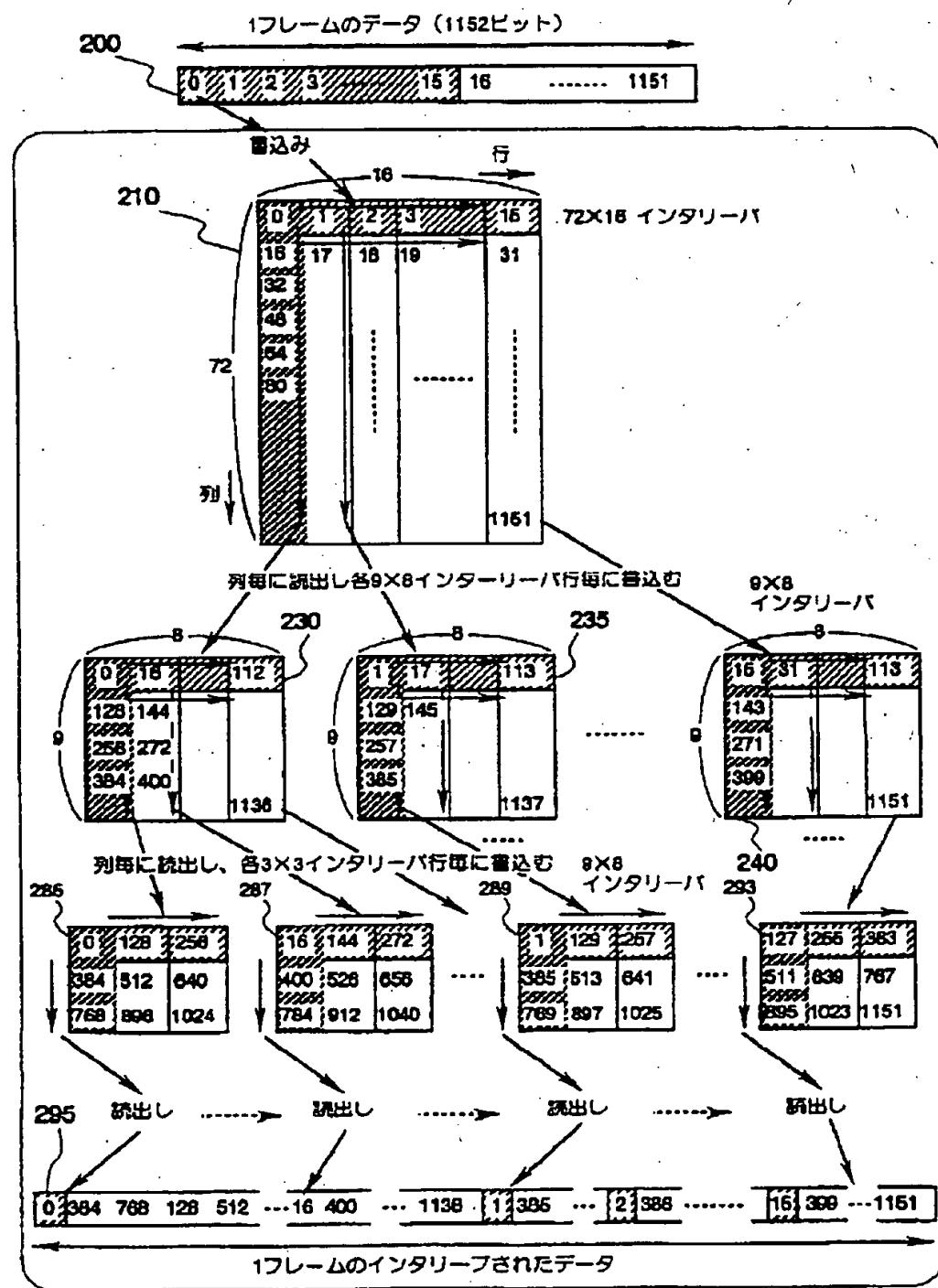
【図2】

FIG. 2



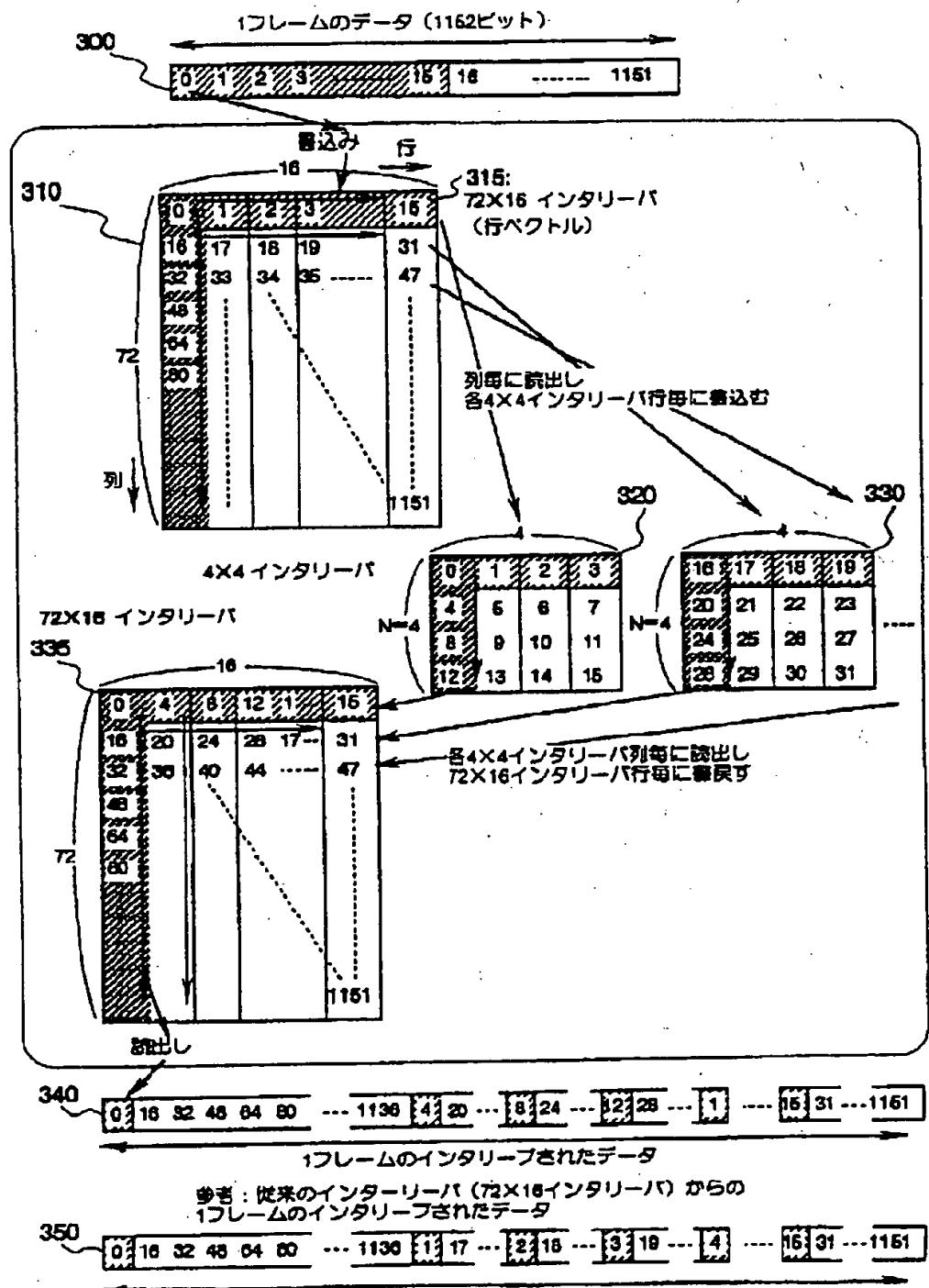
【図3】

FIG. 3



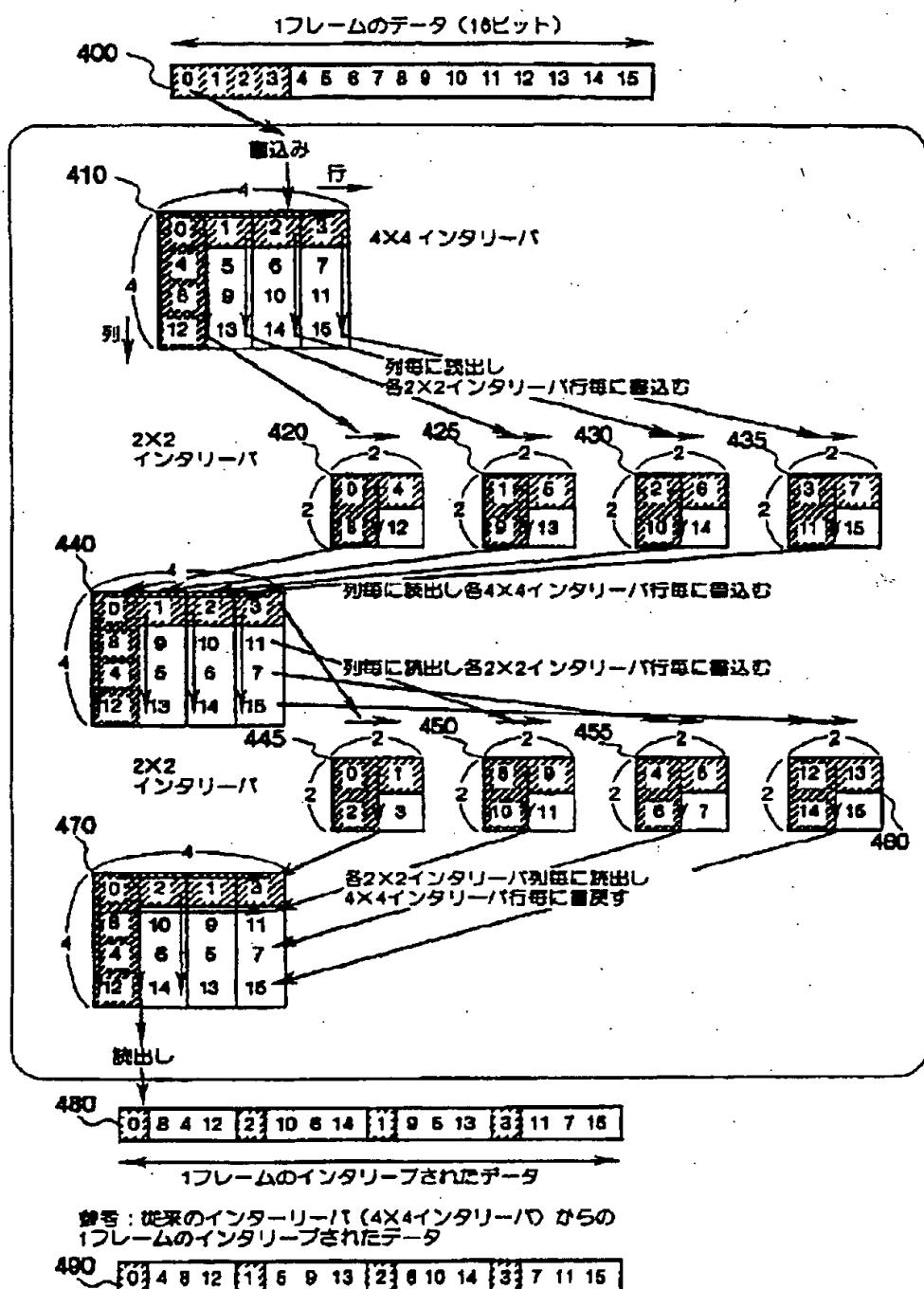
【図4】

FIG. 4



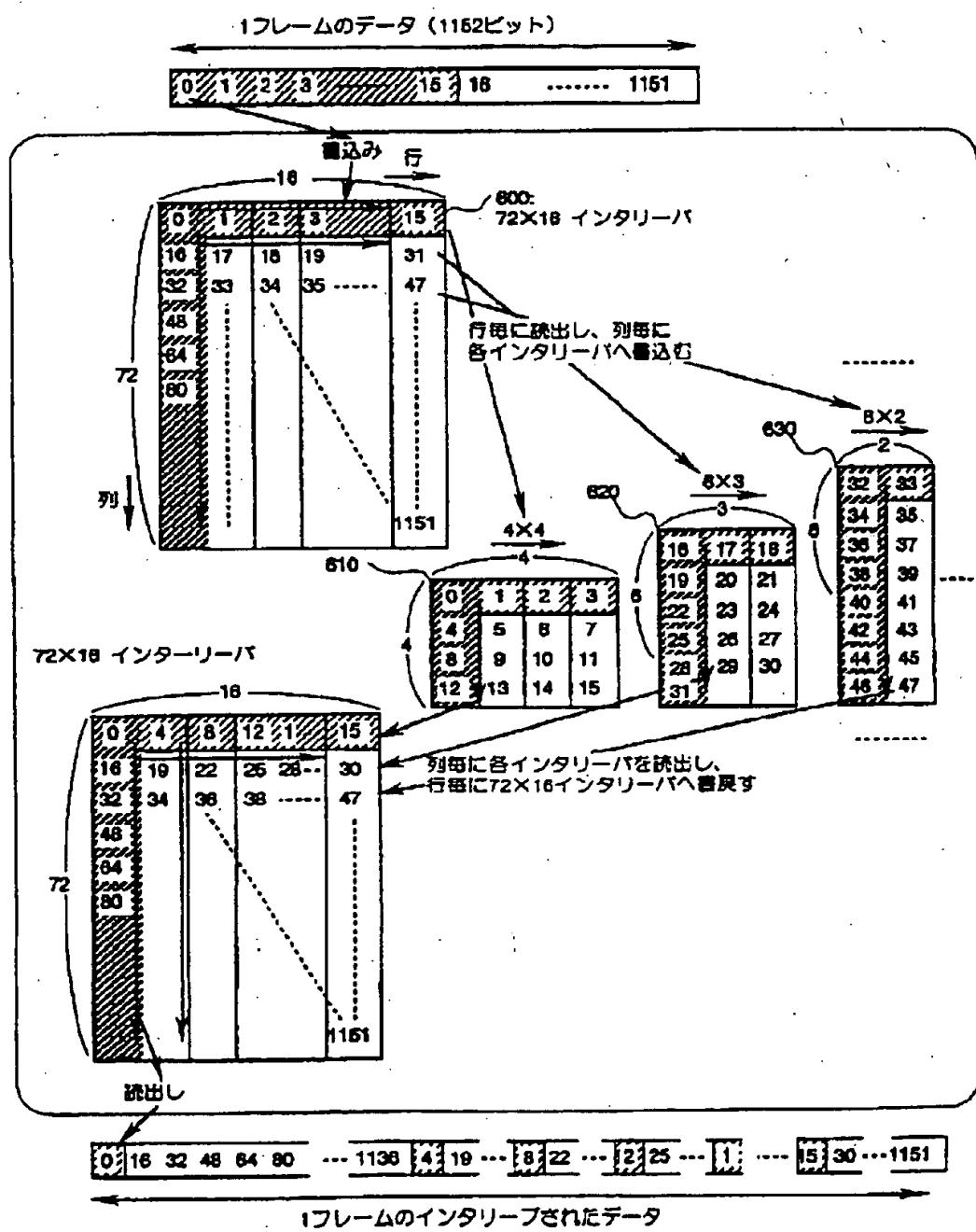
【図5】

FIG. 5



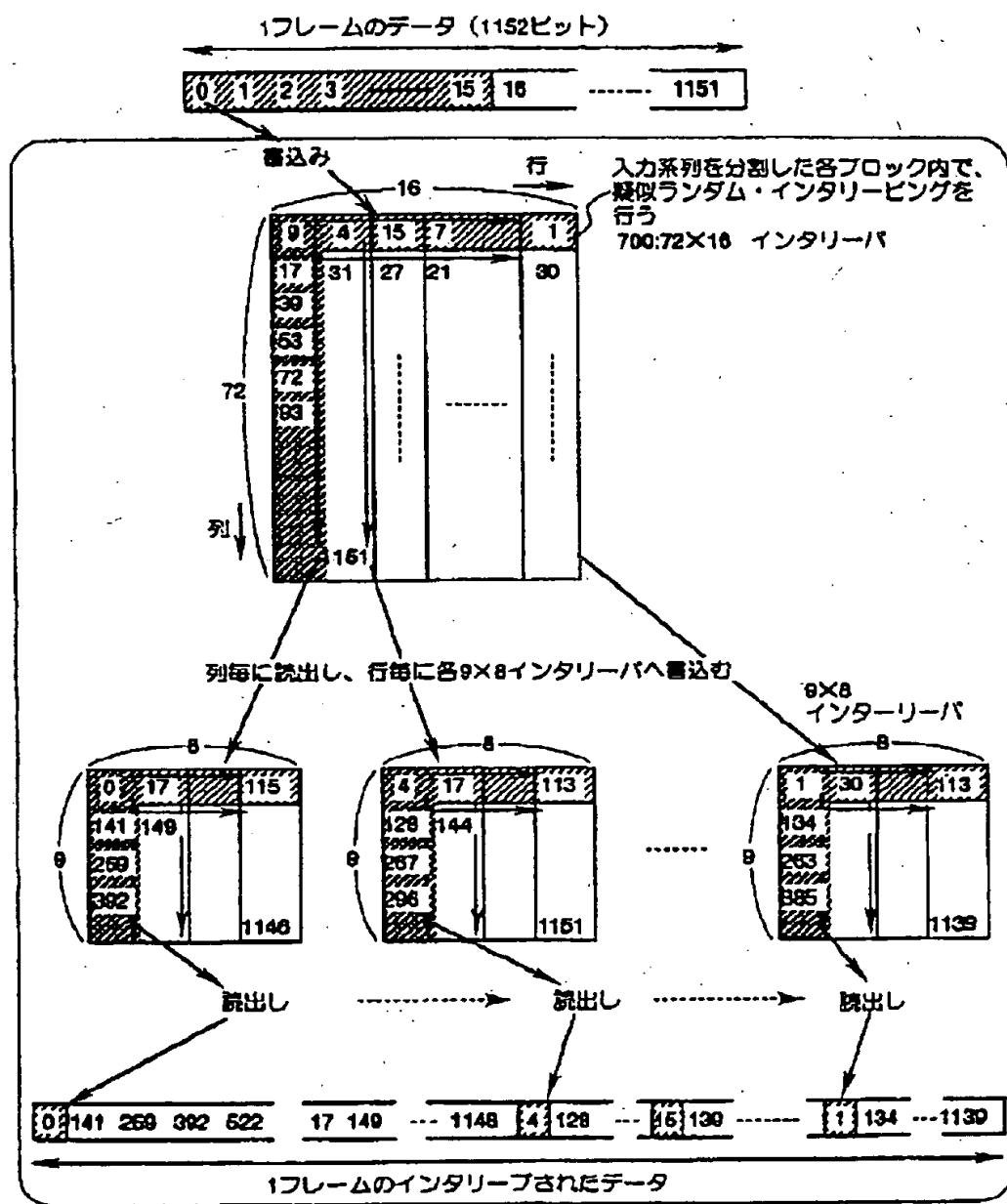
【図6】

FIG. 6



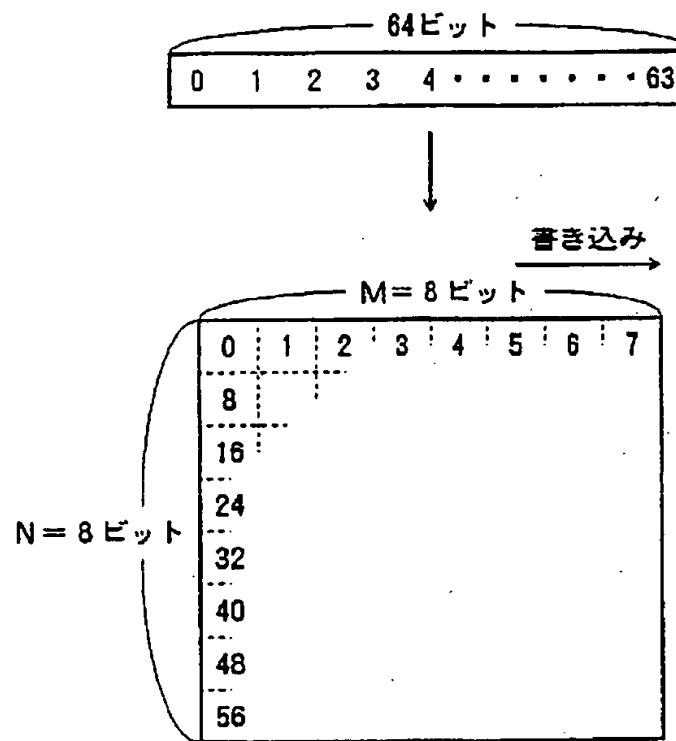
【図7】

FIG. 7



【図8】

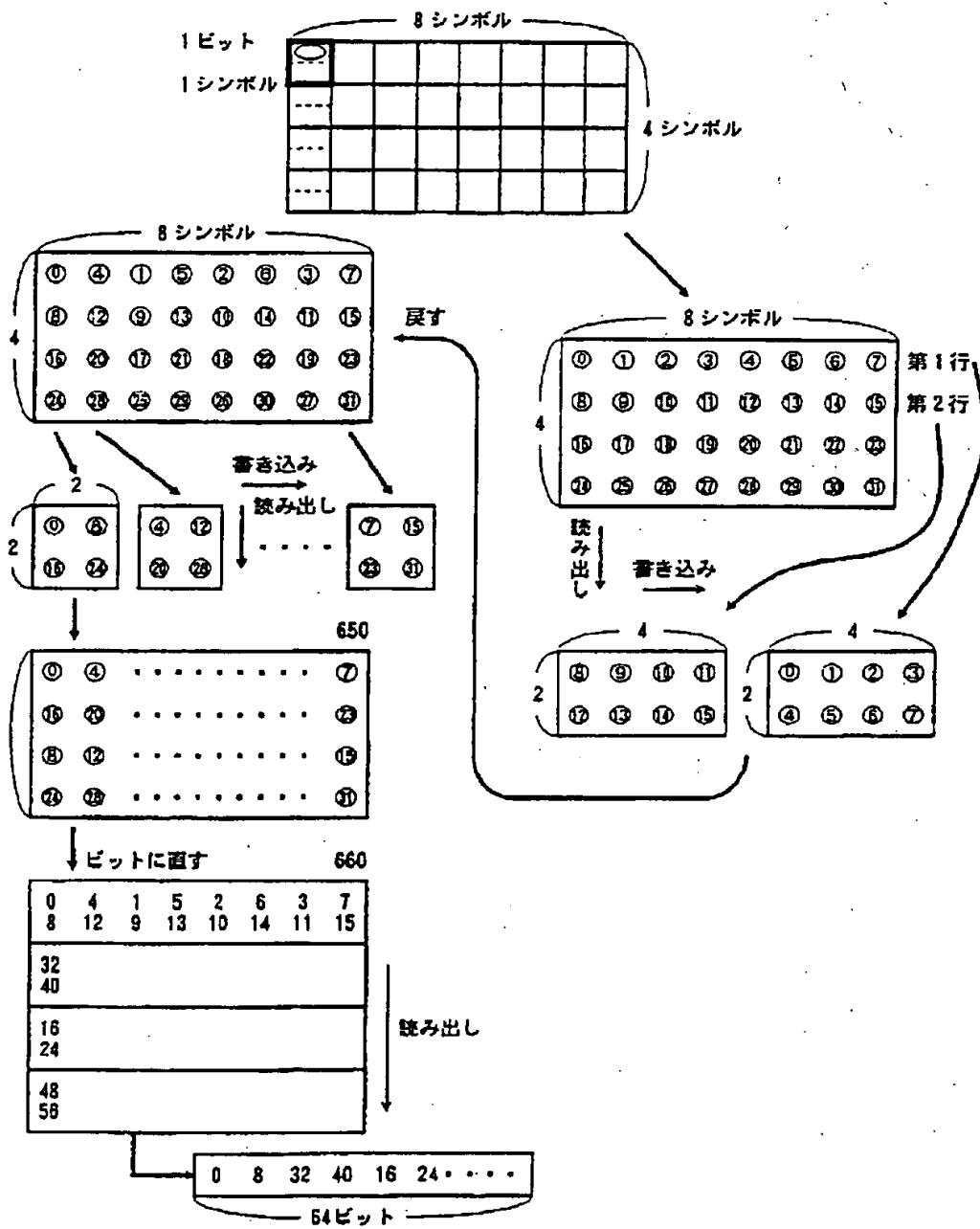
## FIG. 8



【図9】

## FIG. 9

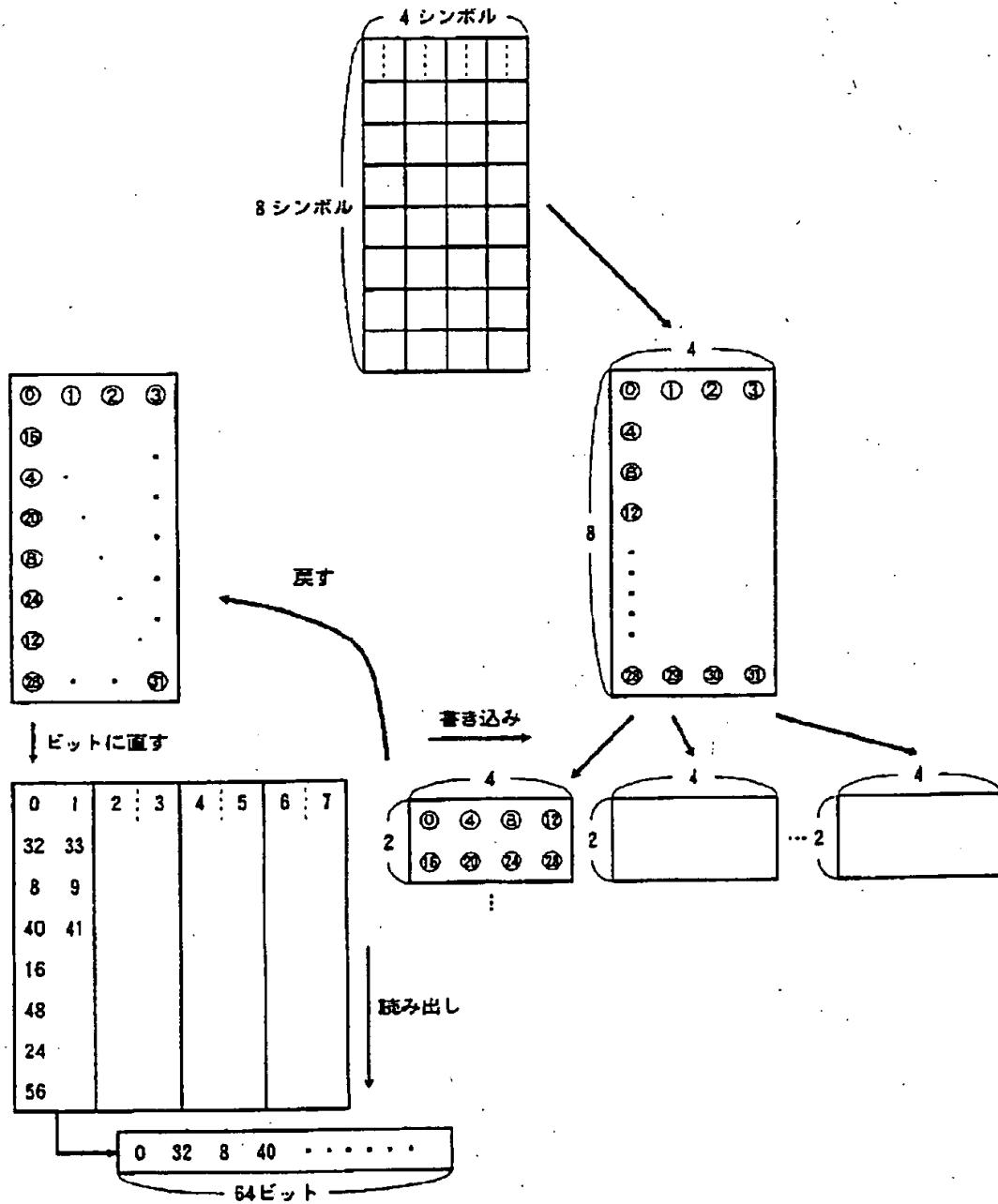
K = 2ビット 1シンボルの場合図に運ぶ



【図10】

## FIG. 10

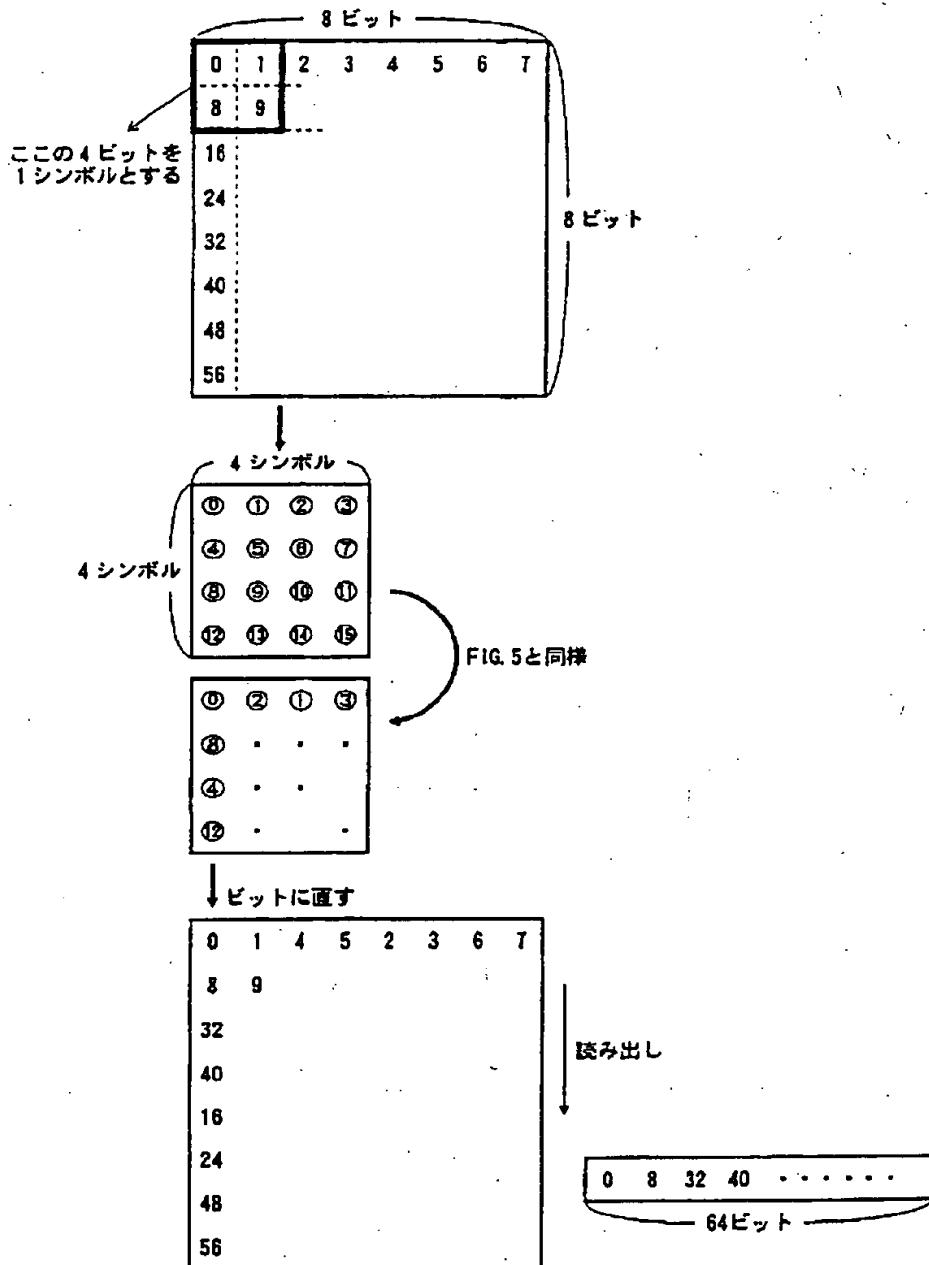
K = 2ビット 1シンボルの場合機に連続



【図11】

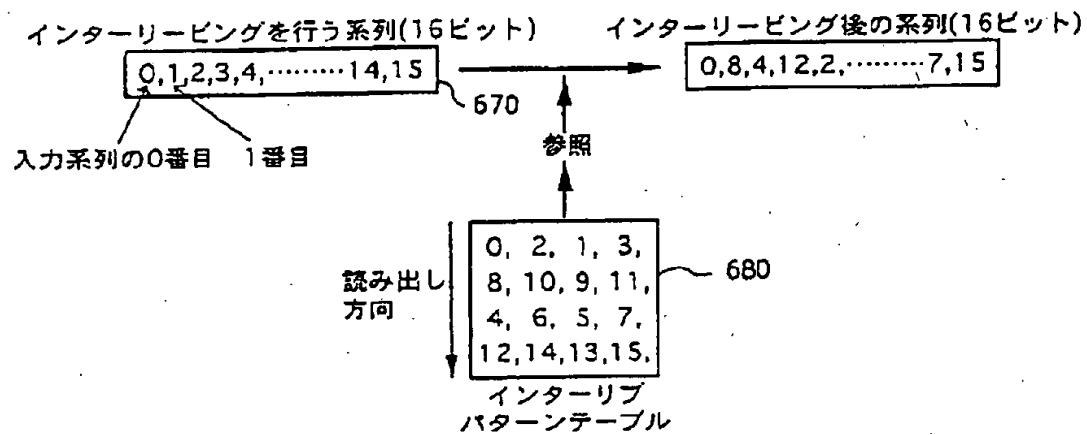
## FIG. 11

・K=4ビット 1シンボル 隣接する4ビットを1シンボルとする場合



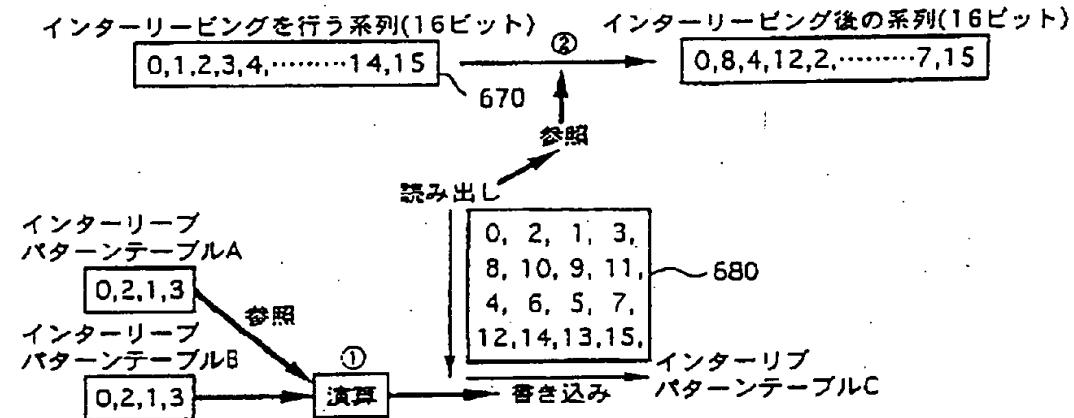
【図12】

FIG. 12



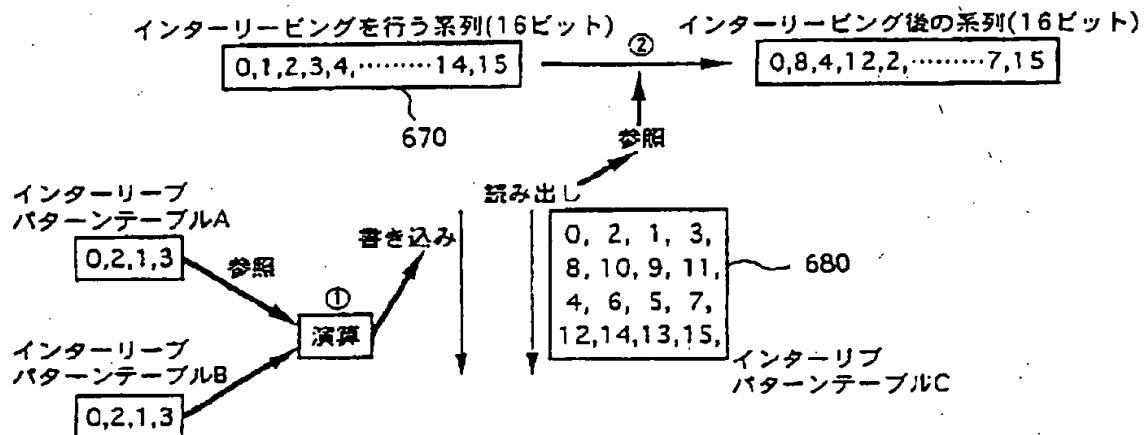
【図13】

FIG. 13



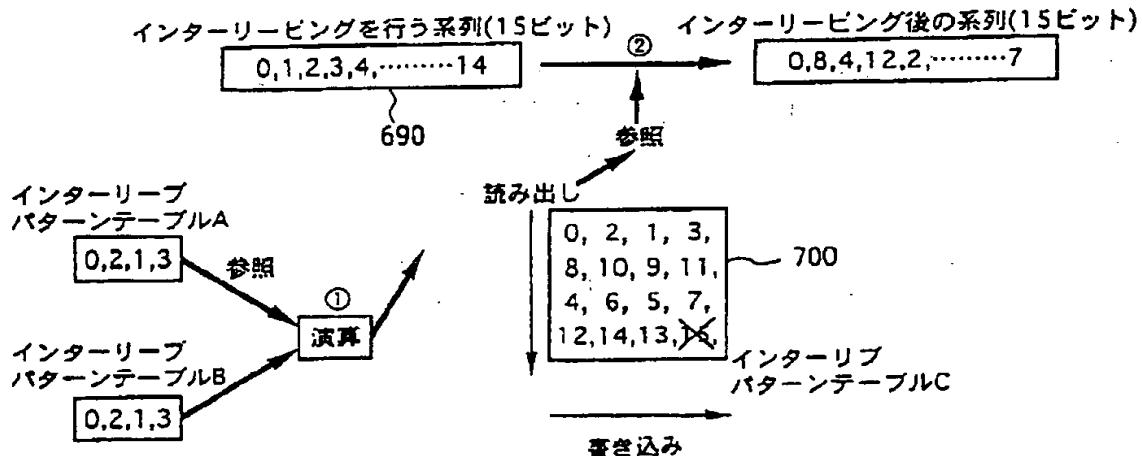
【図14】

FIG. 14



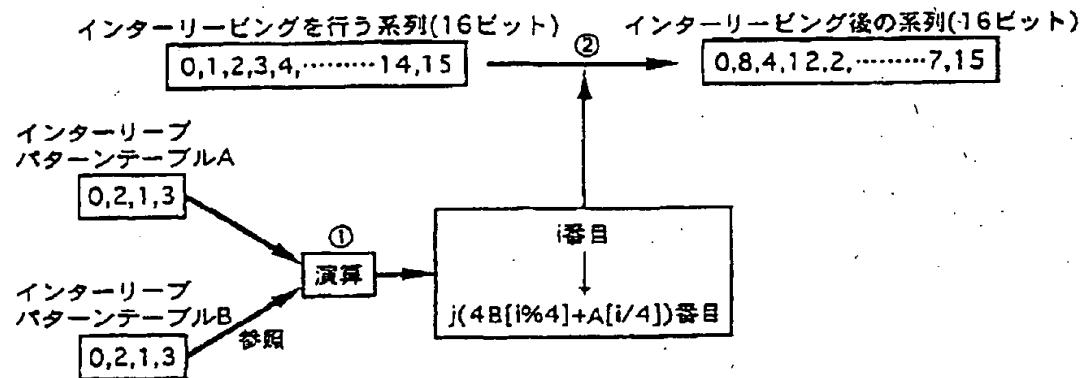
【図15】

FIG. 15



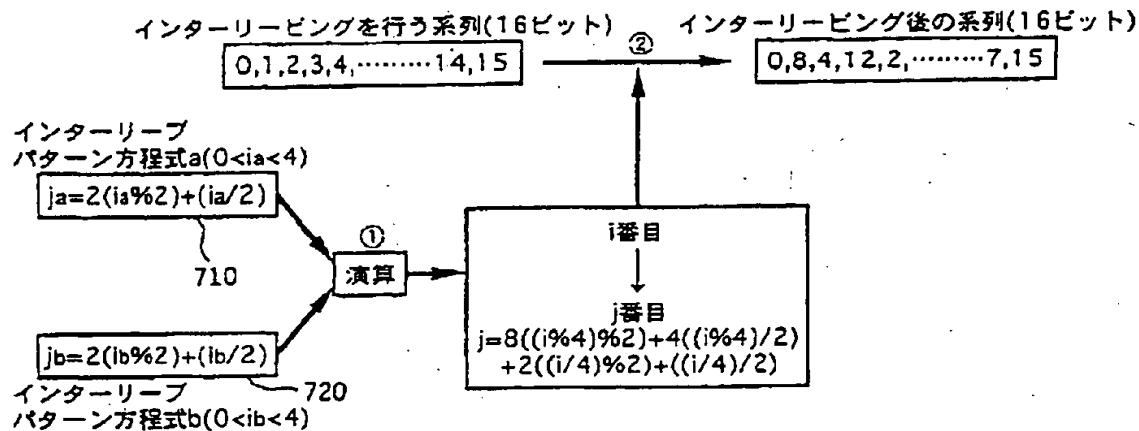
【図16】

FIG. 16



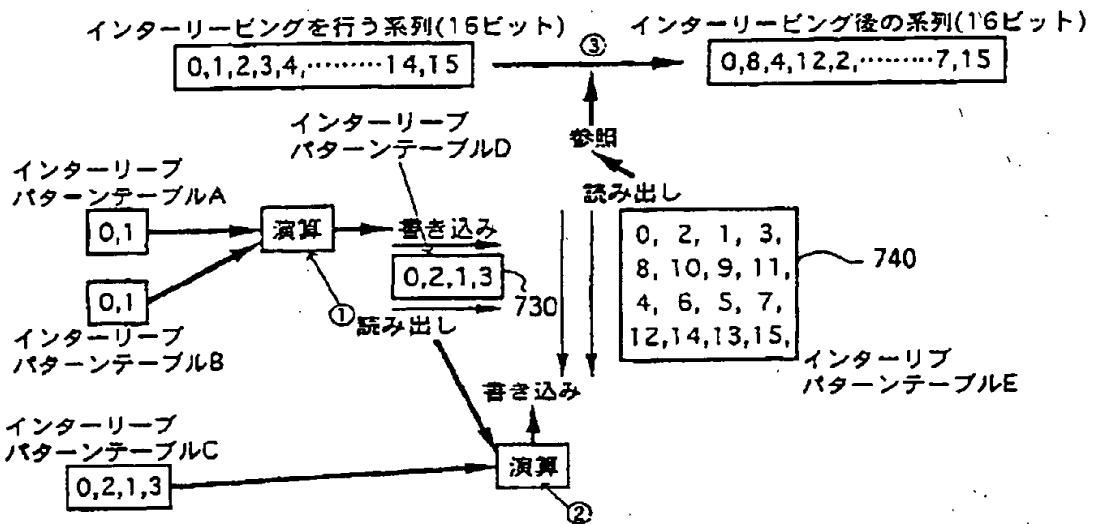
【図17】

FIG. 17



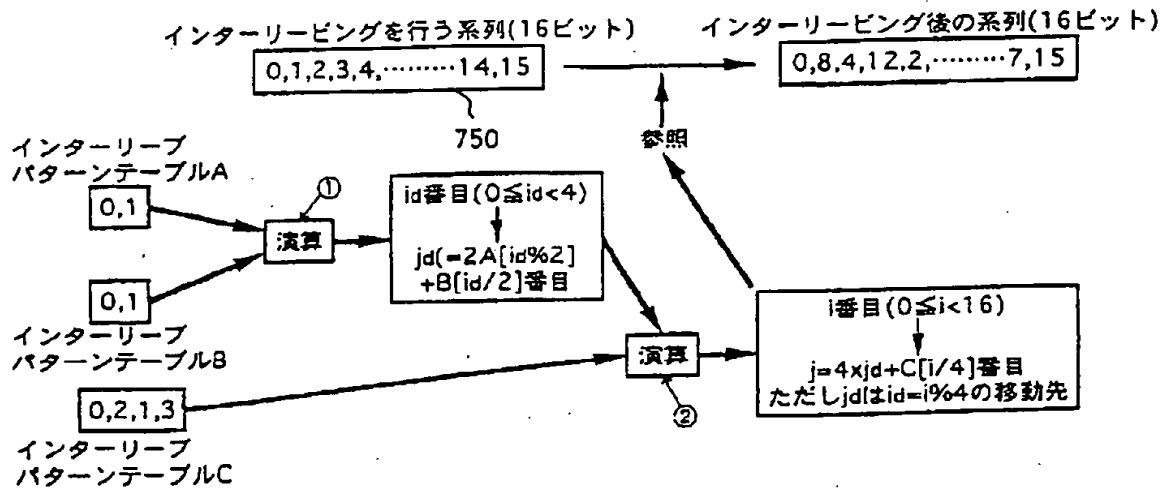
【図18】

FIG. 18



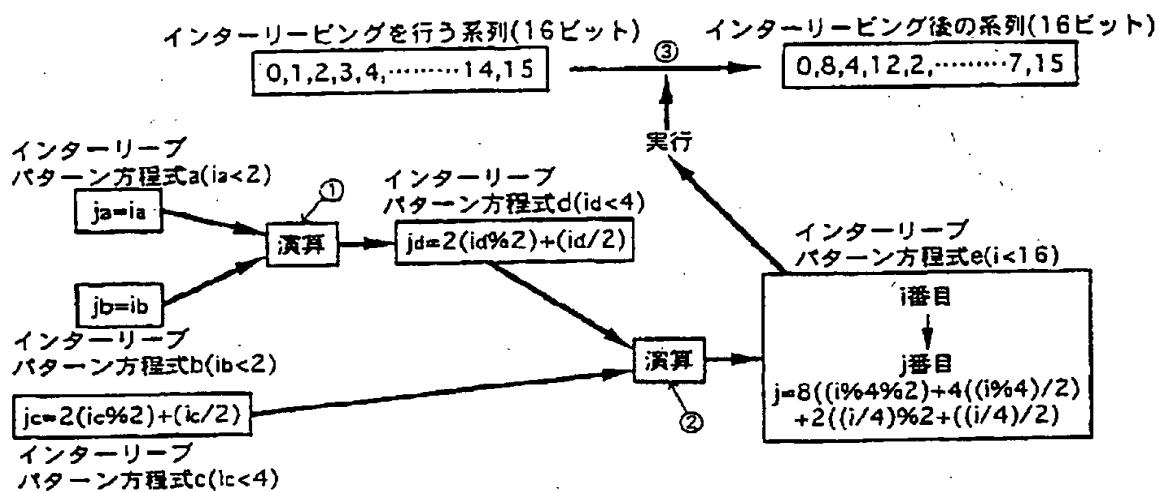
【図19】

FIG. 19



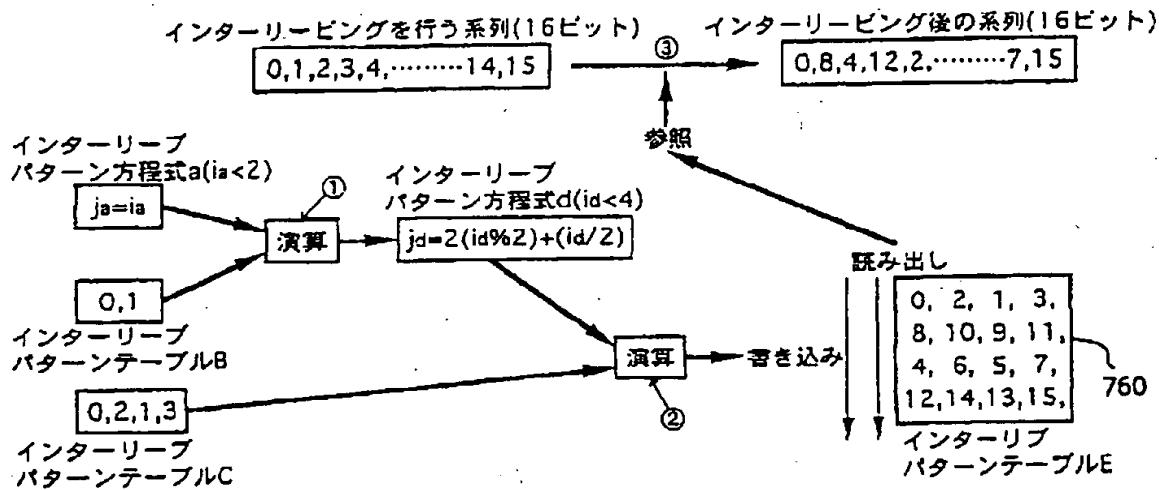
【図20】

FIG. 20



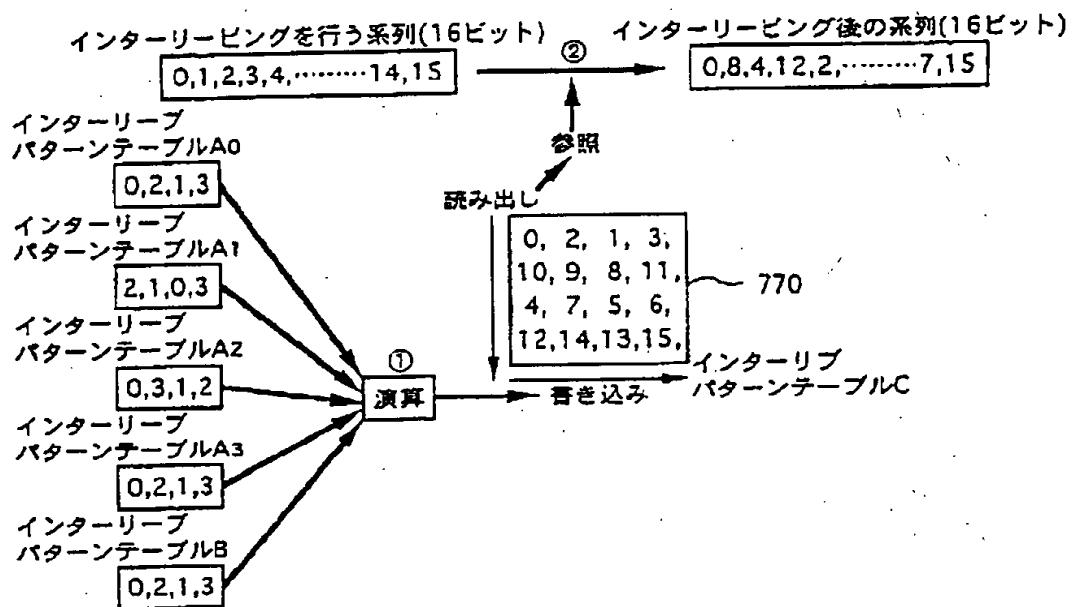
【図21】

FIG. 21



【図22】

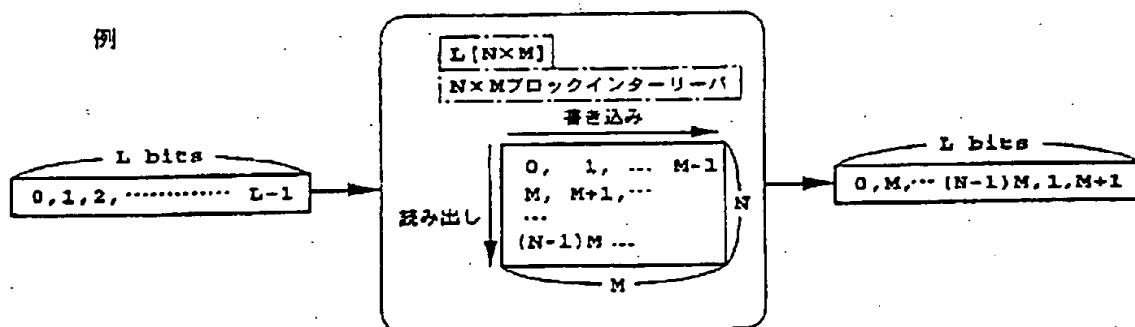
FIG. 22



【図23】

FIG. 23

定義1;  $L[N \times M] \dots N \times M$  ブロックインターリーバで以下のようにインターリービングを行う



【図24】

## FIG. 24

定義2;  $R\{A\} \dots$  逆順に並び替える (Aビット)

例

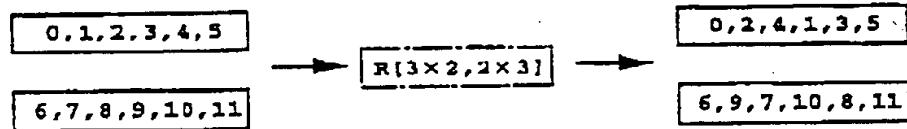


【図25】

## FIG. 25

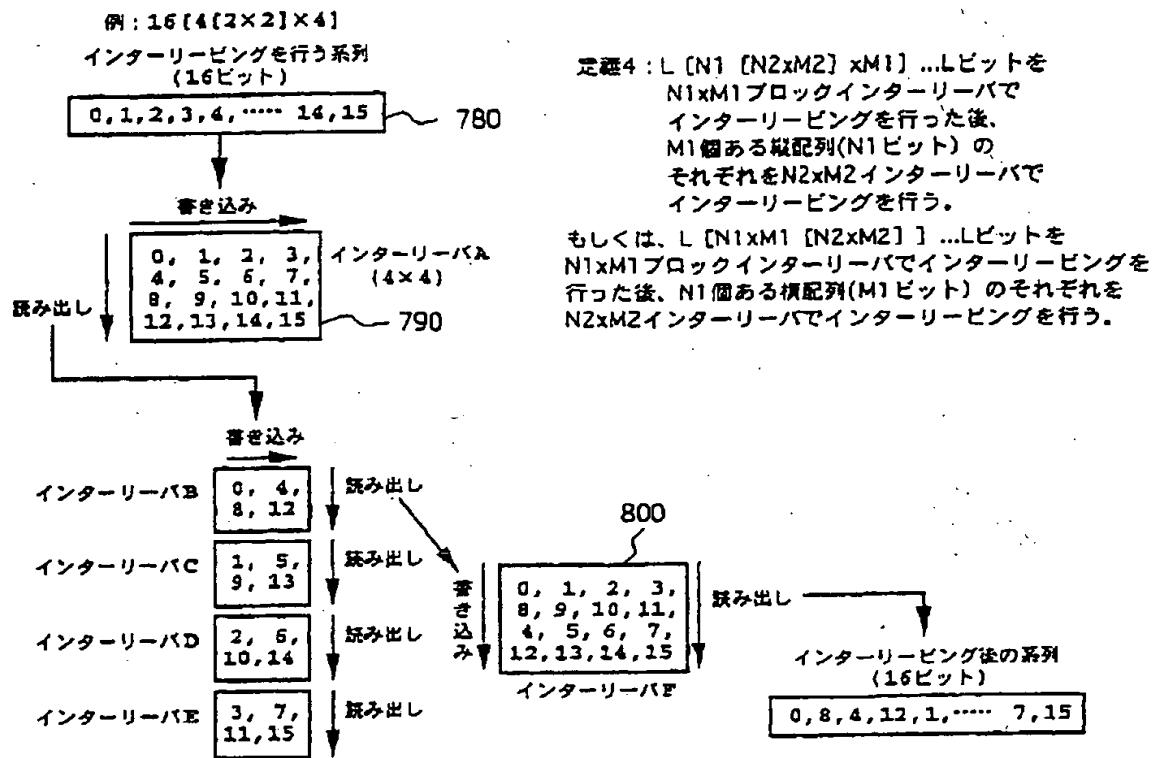
定義3;  $L[N_1 \times M_1, N_2 \times M_2, \dots]$  様数の系列 (各系列はエビット) をそれぞれ  
対応するインターリーバでインターリーピングを行う

例



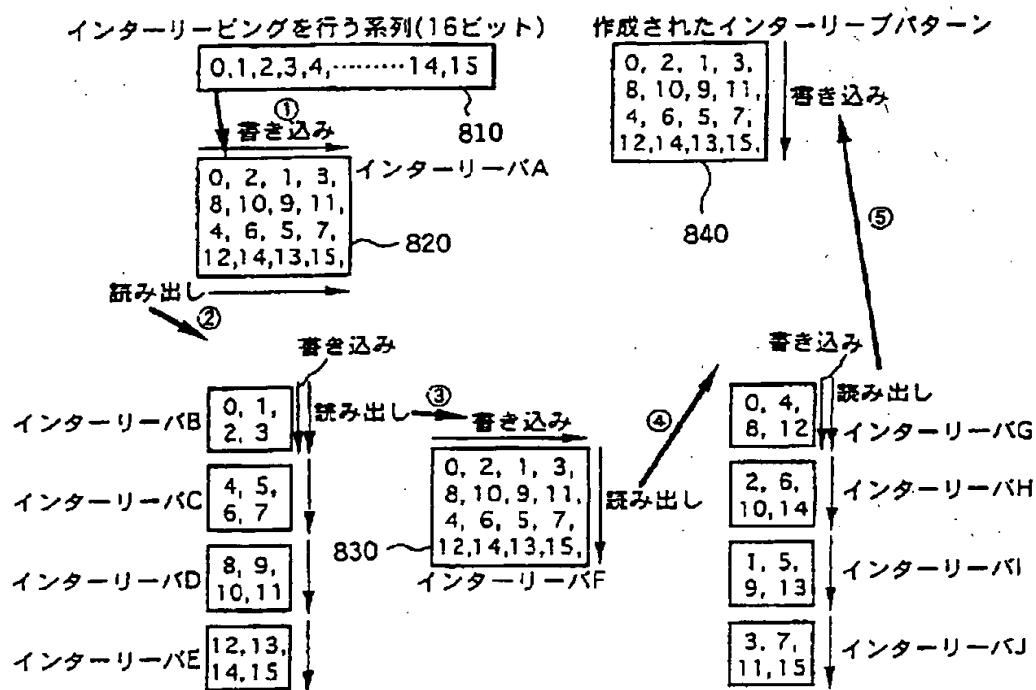
【图 26】

FIG. 26



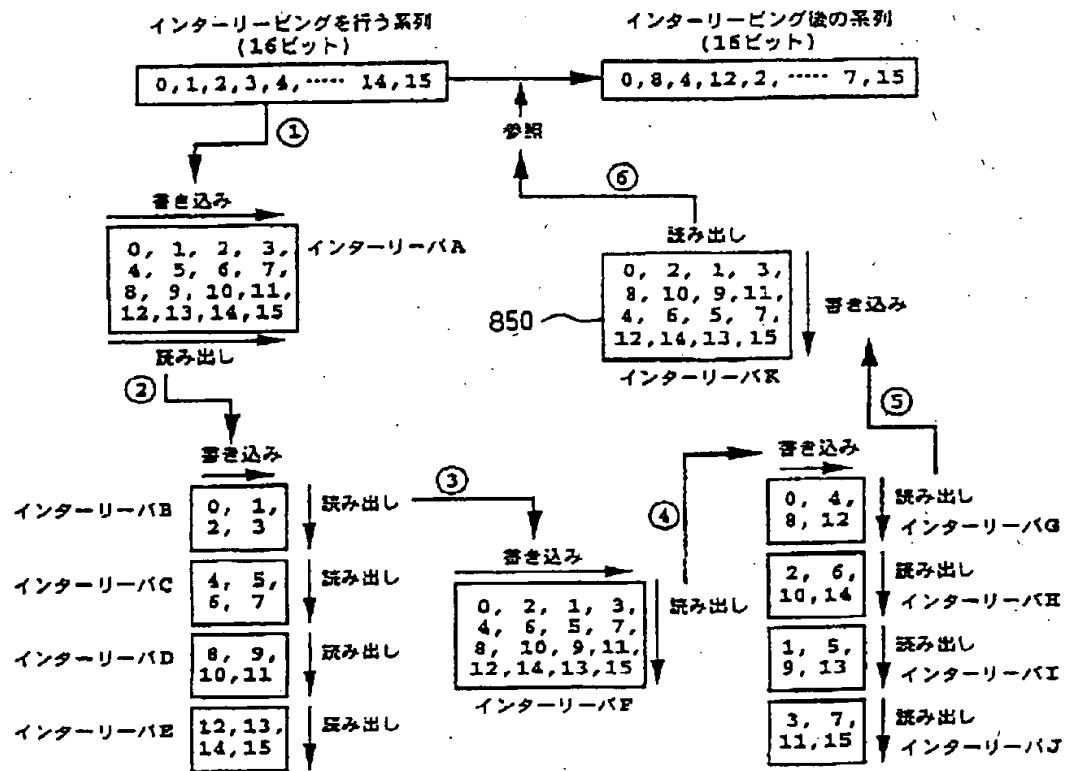
【図27】

FIG. 27



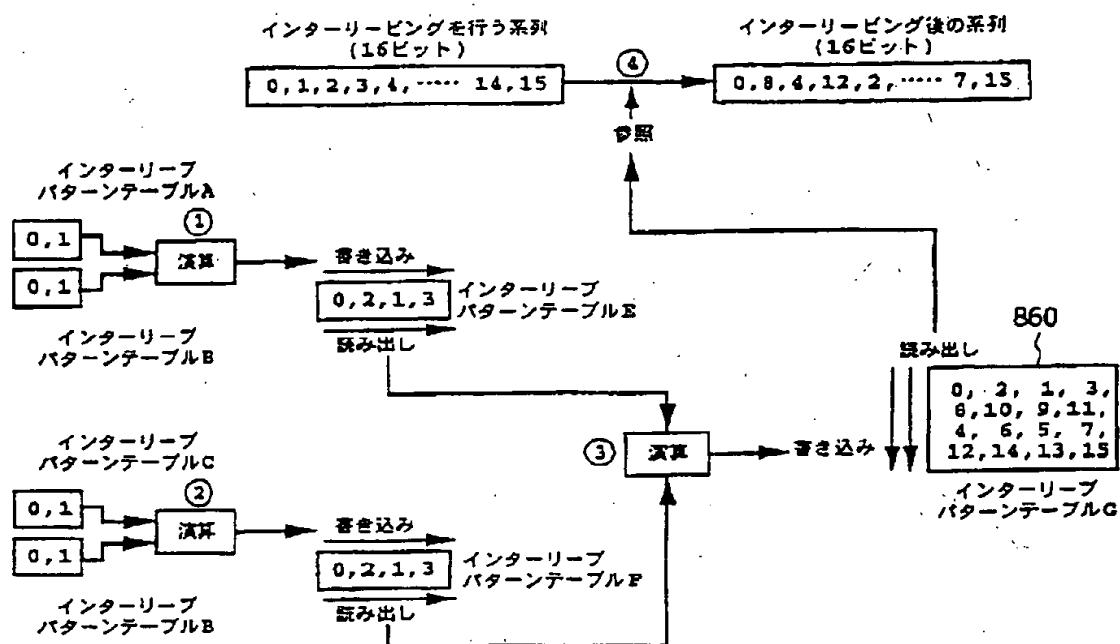
[图28]

FIG. 28



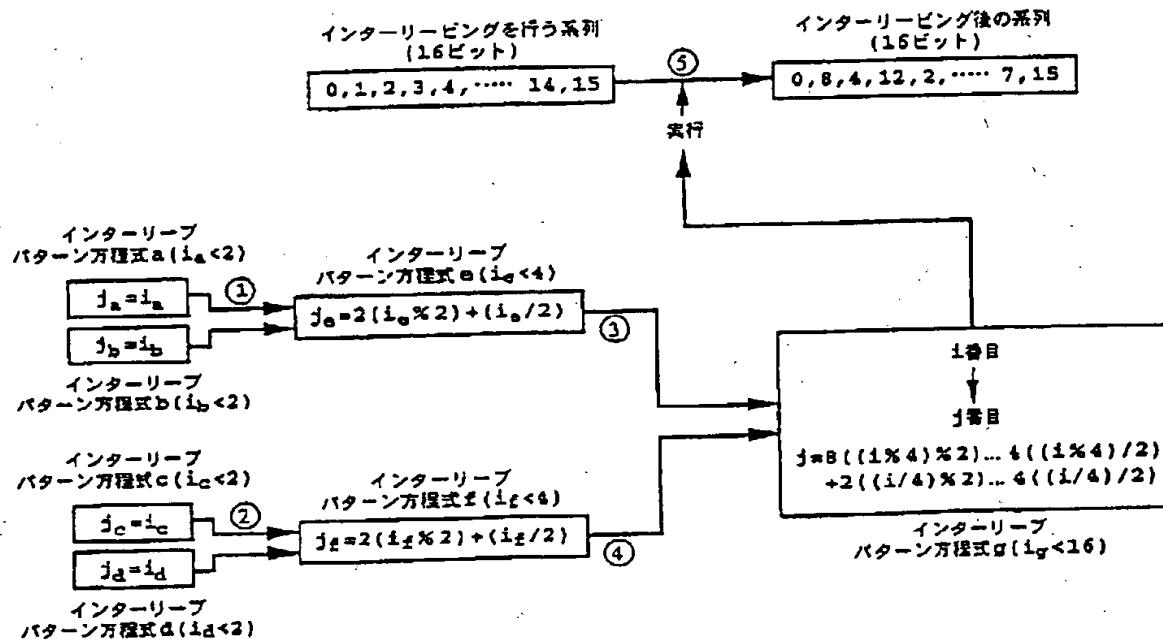
【図29】

FIG. 29



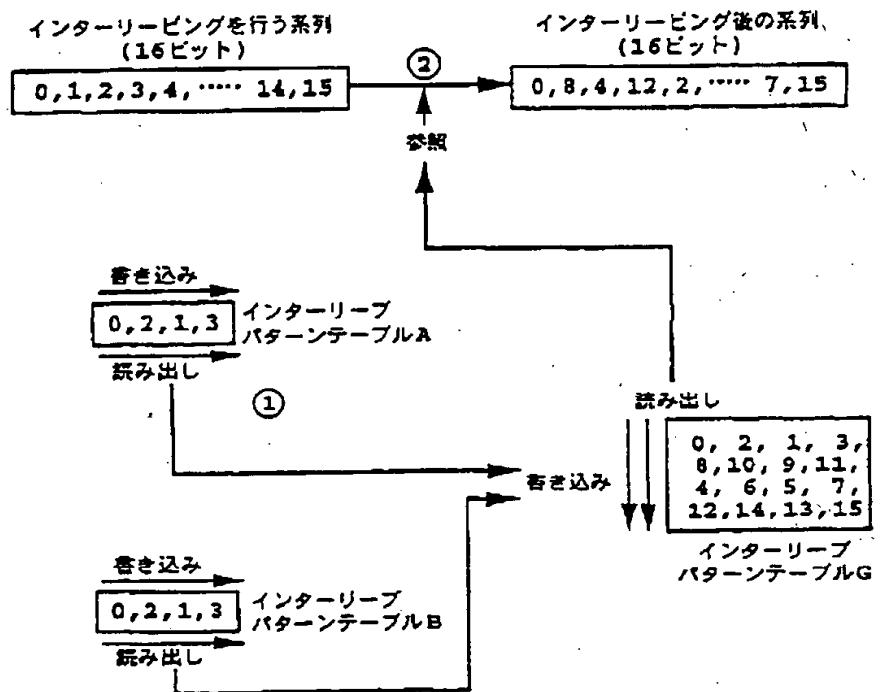
【図30】

FIG. 30



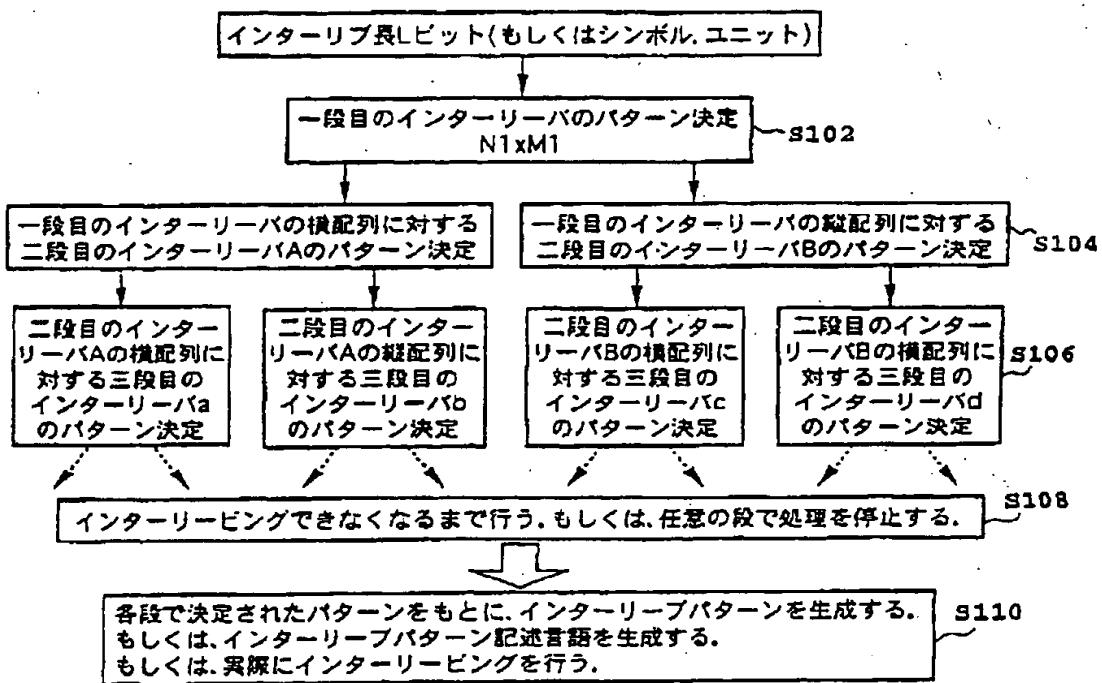
【図31】

FIG. 31



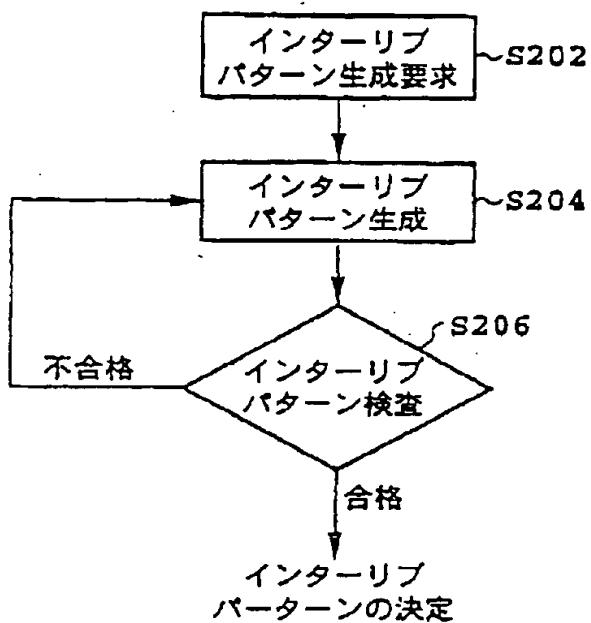
【図32】

FIG.32



【図 33】

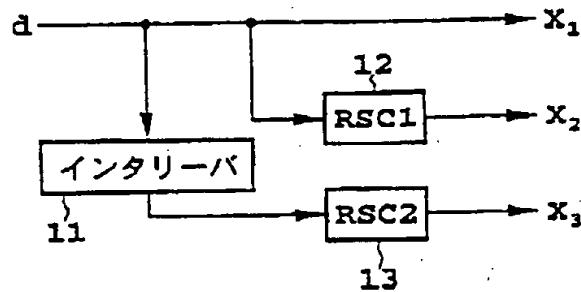
FIG. 33



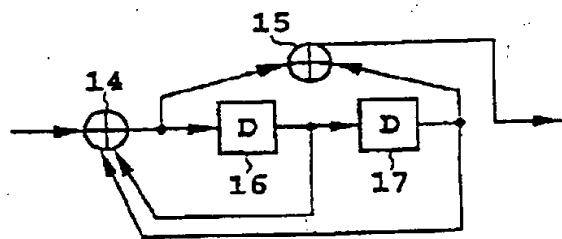
【図34】

FIG. 34

(a)

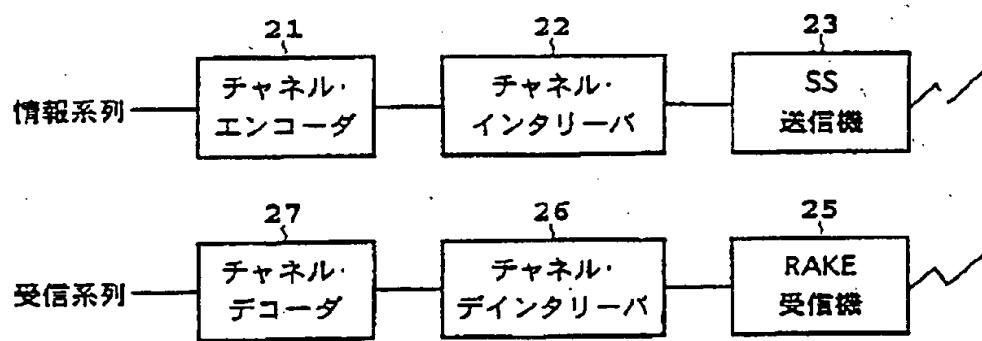


(b)



【図35】

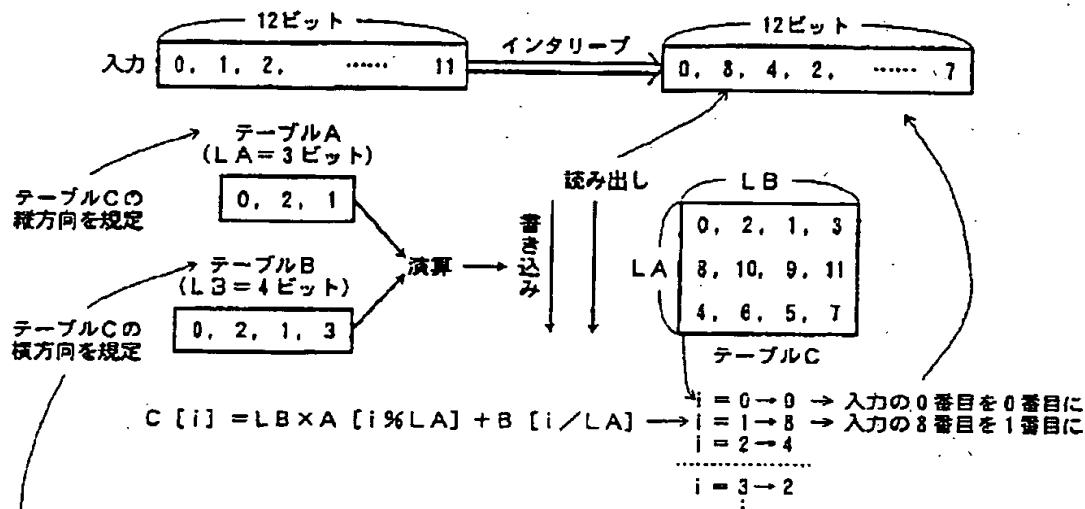
FIG. 35



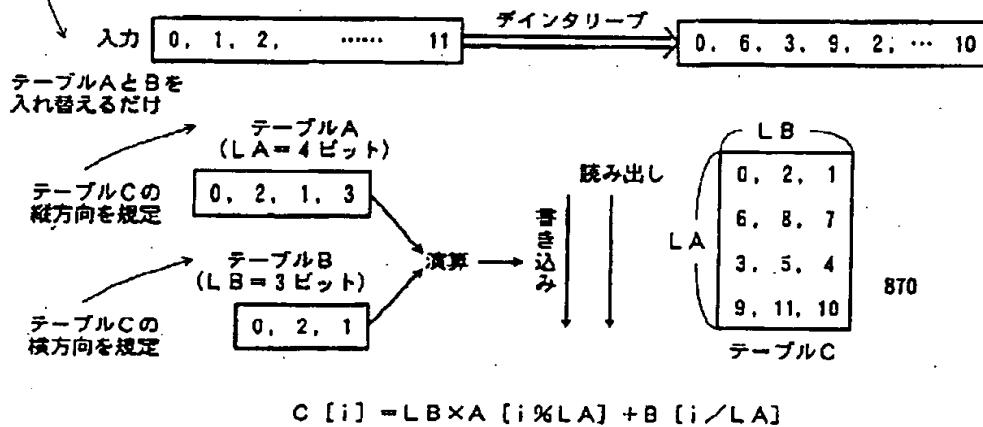
【図36】

## FIG. 36

(a)

• 12ビット (=L) のインタリープ例 ( $L_A \times L_B = 12$ )

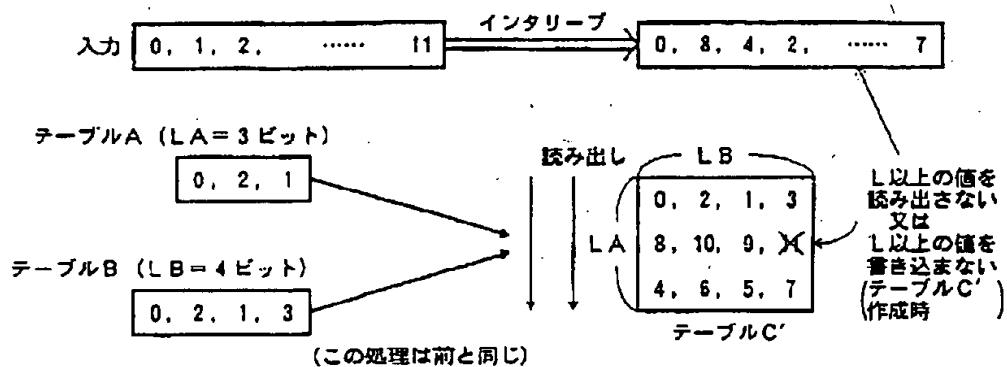
(b)

• 12ビット (=L) のデインターリープ例 ( $L_A \times L_B = 12$ )

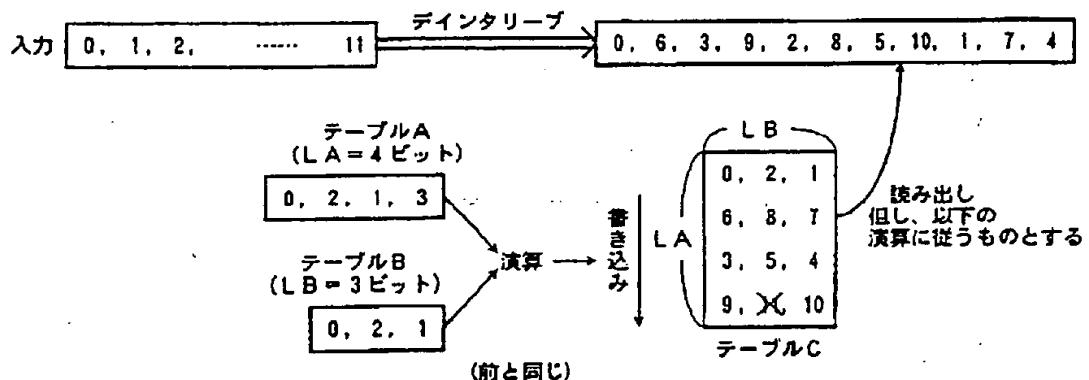
【図37】

## FIG. 37

(a)

・ 11ビット (=L) のインタリーブ例 ( $L \times LB > L$  の場合)

(b)

・ 11ビット (=L) のデインタリーブ例 ( $L \times LB > L$ )

$$C[i] = LB \times A[i \% LA] + B[i / LA] - \alpha$$

$$(0 \leq i \leq (L-1))$$

但し、 $\alpha$ は以下の規則に従う

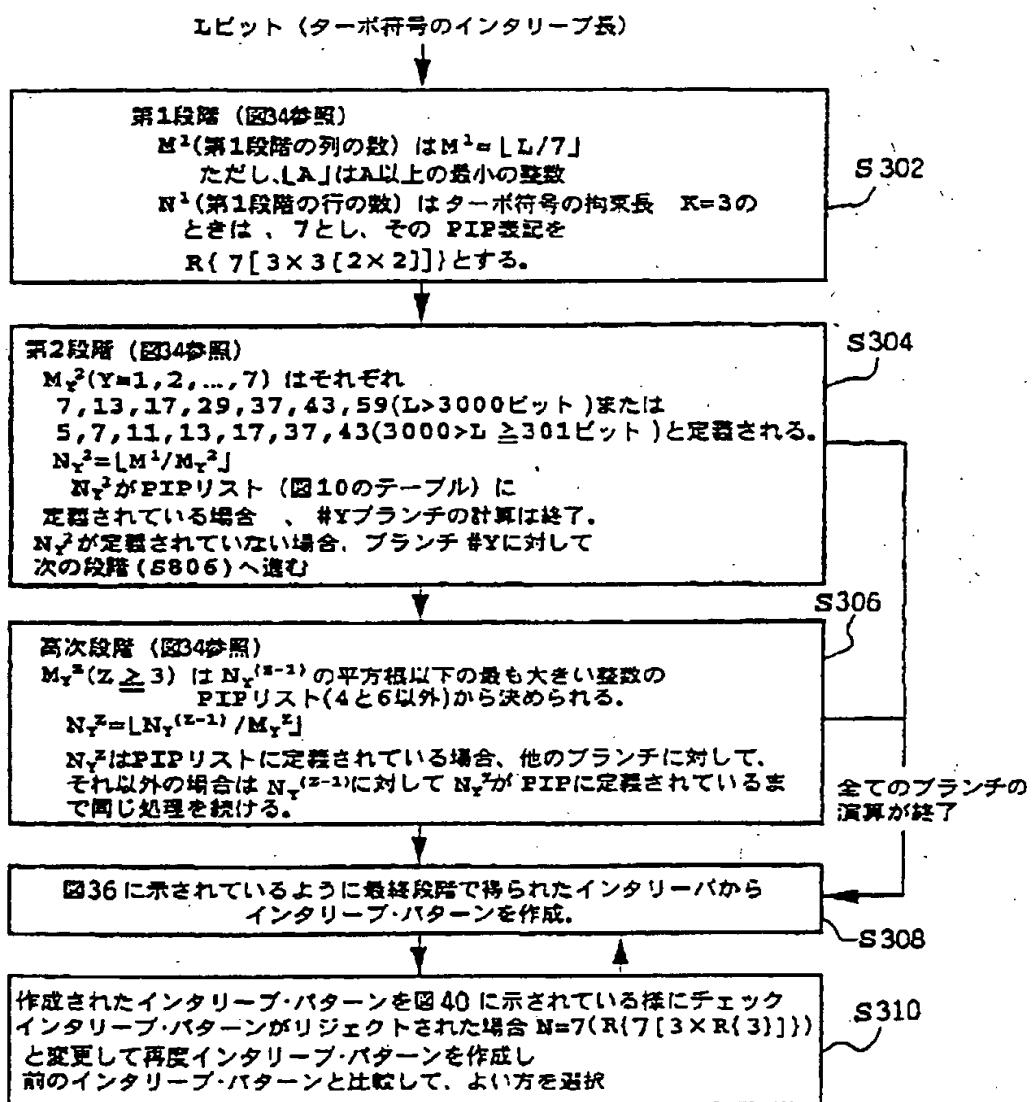
```

 $\alpha = 0$ 
for (j = 0, j < (LAXLB - L), j++) {
if C[i] >= LB * A[(L - 1 - j) % LA]
    + B[(L - 1 - j) / LA]
    alpha++;
}

```

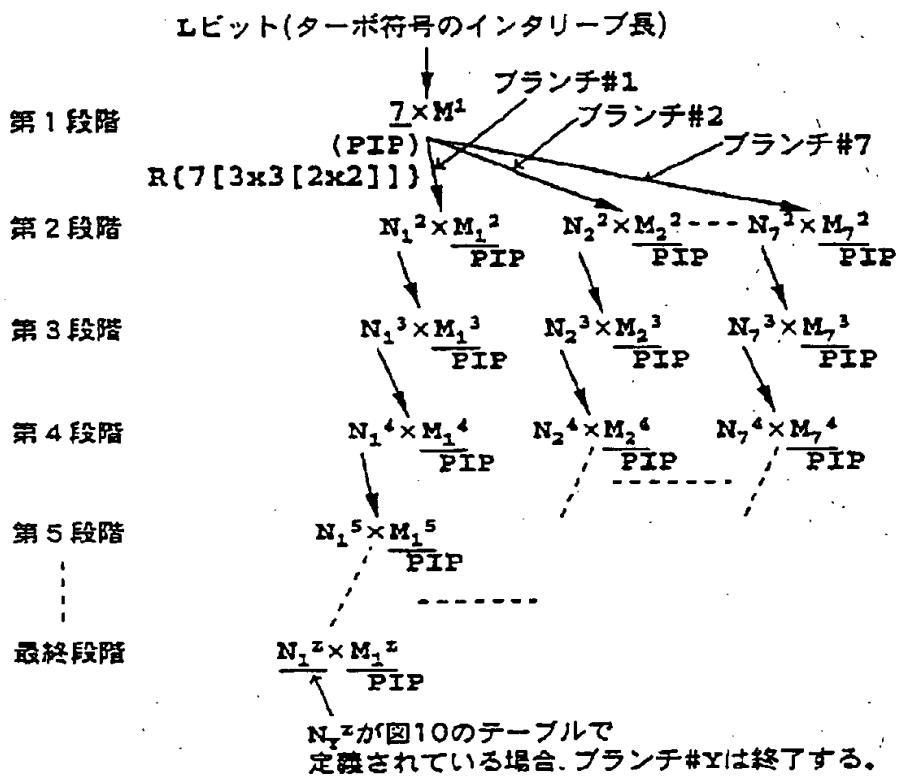
【図38】

FIG. 38



【図39】

FIG. 39



$$L[N_1^1\{PIP\} \times M_1^1[N_1^1[N_1^2[\dots] \times M_1^2\{PIP\}] \times M_1^1\{PIP\}], \\ N_2^1[\dots] \times M_2^1\{PIP\}, \dots, N_7^1[\dots] \times M_7^1\{PIP\}]$$

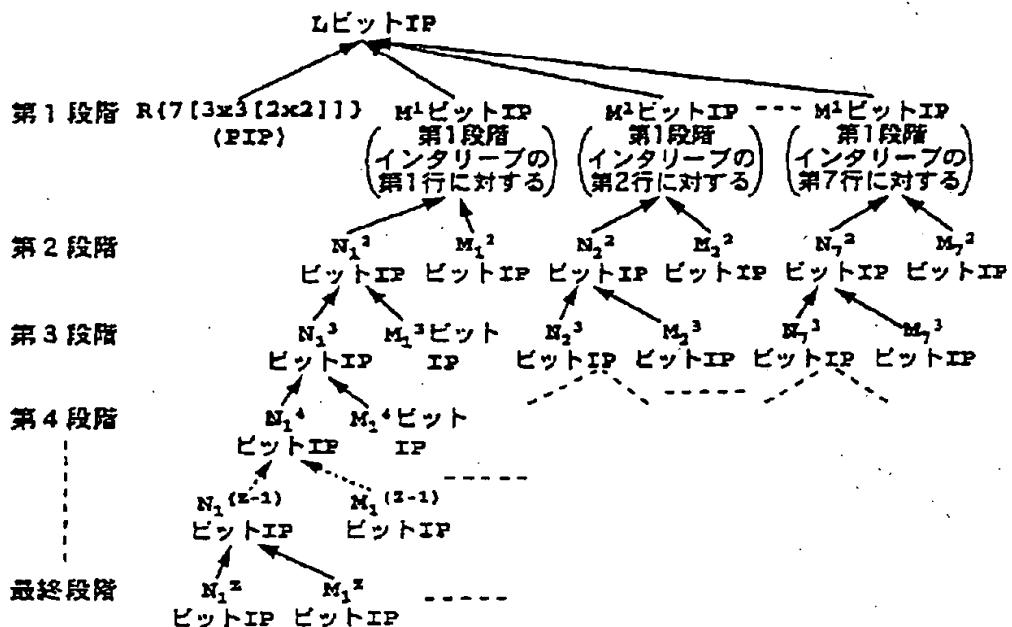
[図40]

FIG. 40

index	表記	インターブ・パターン
T7	R{7[3×3[2×2]]}	4,1,5,2,6,3,0
T7'	R{7[3×R{3}]} 2 R{2}	6,3,0,4,1,5,2 1,0
3	R{3[2×2]}	1,2,0
4	4[2×R{2}]	1,3,0,2
5	5[2×3]	0,3,1,4,2
6	6[3×2]	0,2,4,1,3,5
7	7[3×3[2×2]]	0,3,6,2,5,1,4
8	8[4[2×2]×2]	0,4,2,6,1,5,3,7
9	9[R{2}×5[2×3]]	5,0,8,3,6,1,4,7,2
11	11[3×5[3×2]]	0,5,10,2,7,4,9,1,6,3,8
13	13[2×7[3×3[2×2]] 3[2×2×R{2}[2×R{2}]] ×3[R{3}×1,R{3}×1]]]	0,9,3,12,6,2,11,5,8,1,10,4,7
17	17[4[2×2,4×1,4×1 4×1]×5[3×2]]	0,10,5,15,2,7,12,4,9,14,1,5,11,16,3,8,13
20	20[4[2×R{2}]] ×5[2×3]]	5,15,0,10,8,18,3,13,6,16,1,11,9,19,4,14,7, 17,2,12
29	29[5[3×2] ×7[3×3[2×2]]]	0,14,28,7,21,3,17,10,24,6,20,13,27,2,16,9, 23,5,19,12,26,1,13,8,22,4,18,11,25
37	37[7[3×3] ×6[3×2]]	0,18,36,6,24,12,30,2,20,8,26,14,32,4,22,10, 28,16,34,1,19,7,25,13,31,3,21,9,27,15,33,5, 23,11,29,17,35
43	63[4[2×2] ×11[3×5[3×2]]]	0,22,11,33,5,27,16,38,10,32,21,2,24,13,35,7, 29,18,40,4,26,15,37,9,31,20,42,1,23,12,34,6, 28,17,39,3,25,14,36,8,30,19,41
47	47[7[3×3]×7[3×3]]	0,21,42,7,28,14,35,3,24,45,10,31,17,38,6,27, 13,34,20,41,1,22,43,8,29,15,36,4,25,46,11, 32,18,39,2,23,44,9,30,16,37,5,26,13,33,19, 40
53	53[5[2×3] ×11[3×5[3×2]]]	0,33,11,44,22,5,38,16,49,27,10,43,21,32,2, 35,13,46,24,7,40,18,51,29,4,37,15,48,26,9, 42,20,31,1,34,12,45,23,6,39,17,50,28,3, 36,14,47,25,8,41,19,52,30
59	59[9[R{2}]] ×5[2×3]] ×7[3×3]]	35,0,56,21,42,7,28,49,14,38,3,24,45,10,31, 52,17,41,6,27,48,13,34,55,20,36,1,57,22,43, 8,29,50,15,39,4,25,46,11,32,53,18,37,2,58, 23,44,9,30,51,16,40,5,26,47,12,33,54,19
61	61[5[2×3] ×13[5[2×3] ×3[2×2]]]	0,39,13,52,26,9,48,22,35,3,42,16,55,29,12, 51,25,38,6,45,19,58,32,2,41,15,54,28,11,50, 24,37,5,44,18,57,31,8,47,21,60,34,1,40,14, 53,27,10,49,23,36,4,43,17,56,30,7,46,20,59, 33

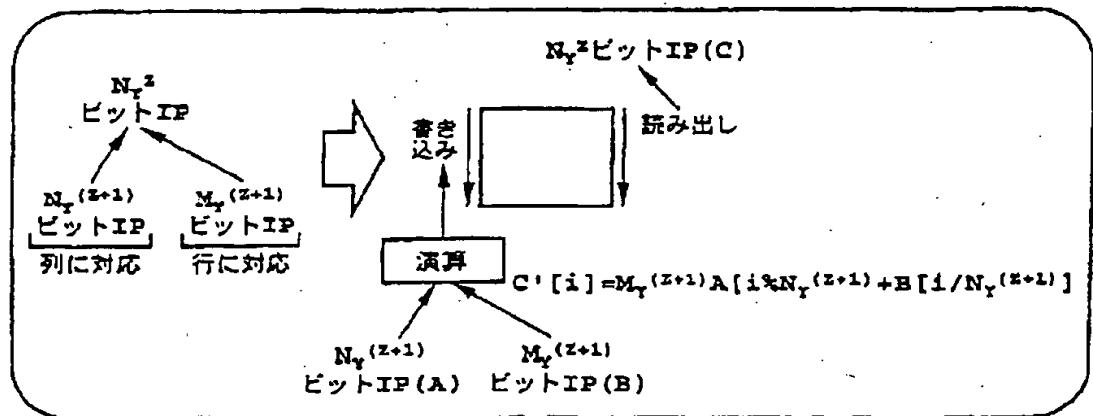
【図41】

FIG. 41



【図42】

FIG. 42

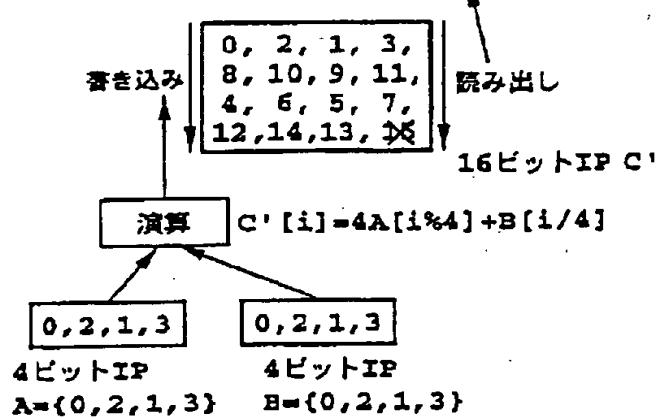


【図43】

FIG. 43

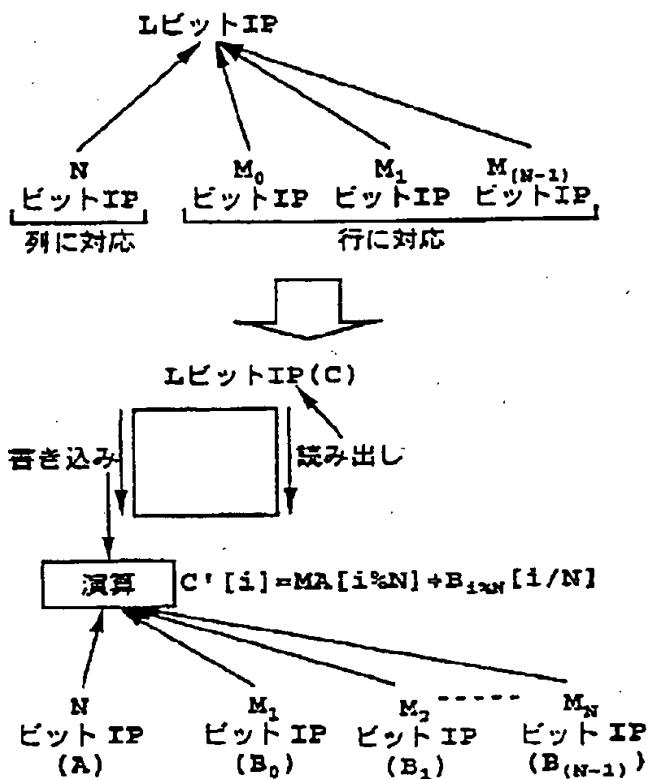
例:

15ビットIP...C={0,8,4,12,2,10,6,14,1,9,5,13,3,11,7}



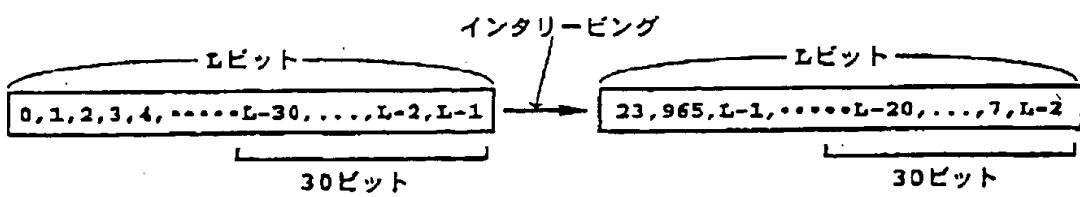
【図44】

FIG. 44



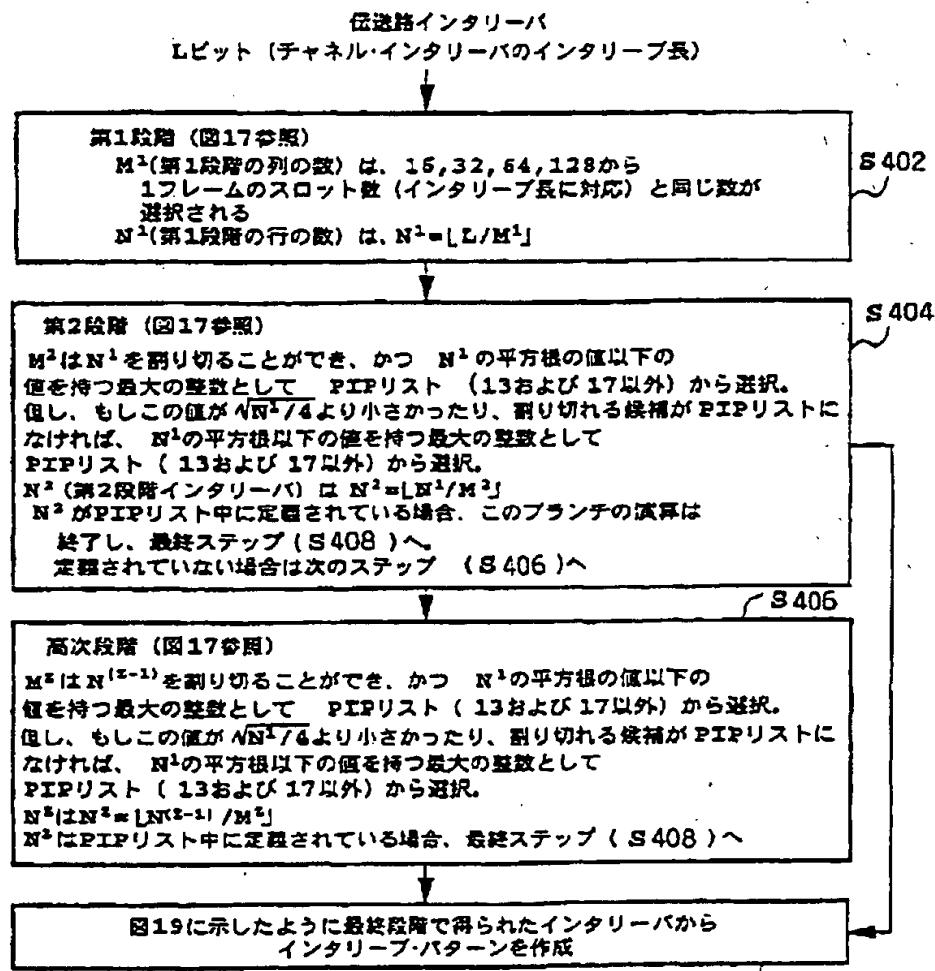
【図45】

FIG. 45



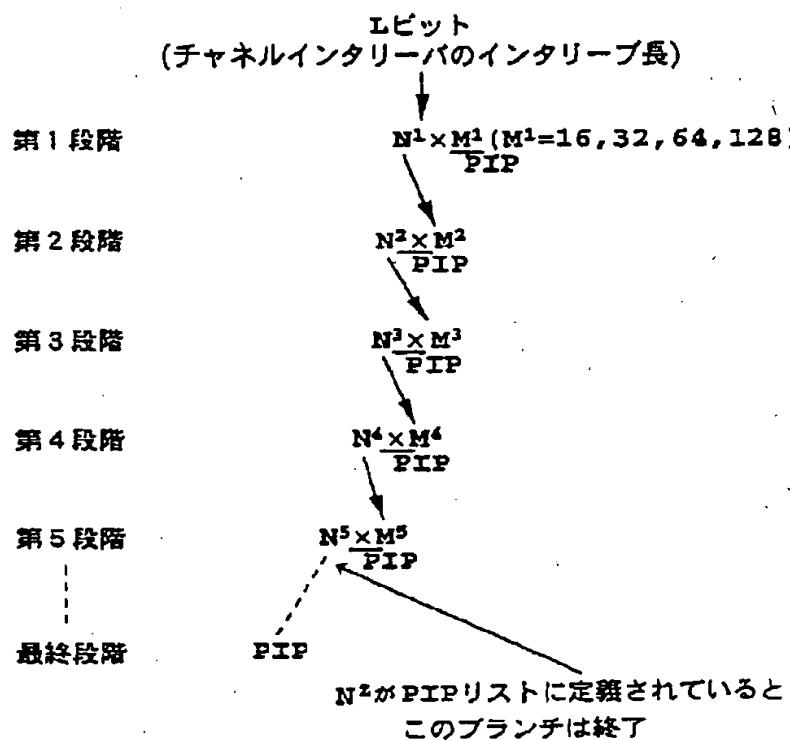
【図46】

## FIG. 46



【図47】

FIG. 47



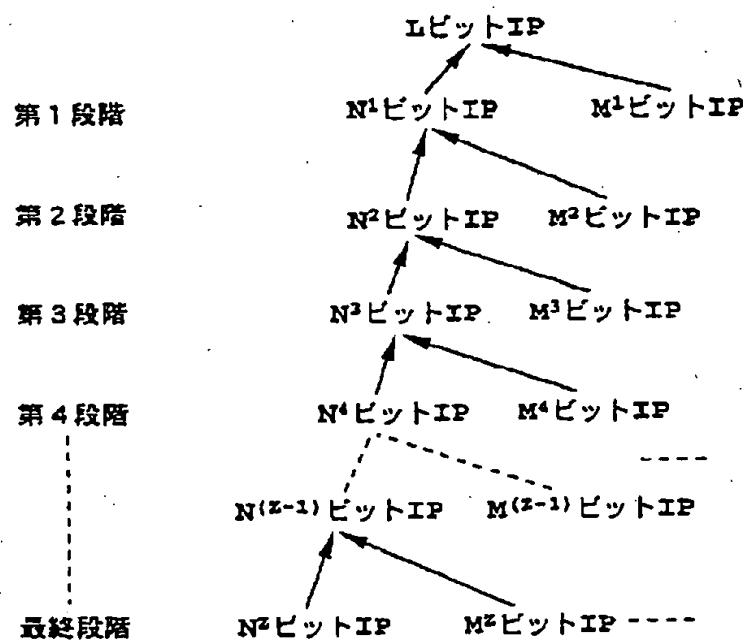
【図48】

FIG. 48

index	表記	インタリーブ・パターン
2	2	0,1
3	3	0,1,2
4	4[2×2]	0,2,1,3
5	5[2×3]	0,3,1,4,2
6	6[3×2]	0,2,4,1,3,5
7	7[3×3[2×2]]	0,3,6,2,5,1,4
8	8[4[2×2]×2]	0,4,2,6,1,5,3,7
9	9[3×3]	0,3,6,1,4,7,2,5,8
10	10[5[3×2]×2]	0,4,8,2,6,1,5,9,3,7
11	11[3×5[3×2]]	0,5,10,2,7,4,9,1,6,3,8
13	13[2×7[3×3[2×2]] ,3[2×2×R{2}2×R{2}] ×3(R{3}×1,R{3}×1,3×1)]	0,9,3,12,6,2,11,5,8,1,10,4,7
16	16[4[2×2] ×4[2×2]]	0,8,4,12,2,10,6,14,1,9,5,13,3,11,7,15
17	17[4[2×2,4×1 ,4×1,4×1,4×1] ×5[3×2]]	0,10,5,15,2,7,12,4,9,14,1,6,11,16,3,8,13
20	20[4[2×2] ×5[3×2]]	0,10,5,15,2,12,7,17,4,14,9,19,1,11,6,16,3, 13,8,18
32	32[8[4[2×2]×2] ×4[2×2]]]	0,16,8,24,4,20,12,28,2,18,10,26,6,22,14,30, 1,17,9,25,5,21,13,29,3,19,11,27,7,23,15,31
64	64[8[4[2×2]×2] ×8[4[2×2]×2]]	0,32,16,48,8,40,24,56,4,36,20,52,12,44,28, 60,2,34,18,50,10,42,26,58,6,38,22,54,14,46, 30,62,1,33,17,49,9,61,25,57,5,37,21,53,13, 45,29,61,3,35,19,51,11,43,27,59,7,39,23,55, 15,47,31,63
128	128[16[4[2×2]× 4[2×2] ]×8[4[2×2]×2]]	0,64,32,96,16,80,48,112,8,72,40,104,24,88, 56,120,4,68,36,100,20,84,52,116,12,76,44, 108,28,92,60,124,2,66,34,98,18,62,50,114,10, 74,42,106,26,30,58,122,6,70,38,102,22,86,56, 118,14,78,46,110,30,94,62,126,1,65,33,97,17, 81,49,113,9,73,41,103,25,89,57,121,5,69,37, 101,21,85,53,117,13,77,45,109,29,93,61,125, 3,67,35,99,19,83,51,115,11,75,43,107,27,91, 59,123,7,71,39,103,23,87,55,119,15,79,47, 111,31,95,63,127

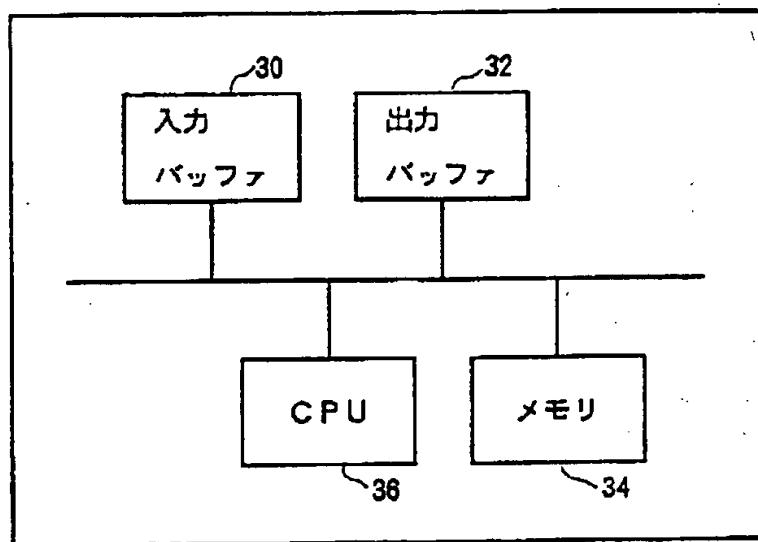
【図49】

FIG. 49



【図50】

FIG. 50



## 【国際調査報告】

国際調査報告		国際出願番号 PCT/JP98/05027										
<p>A. 発明の属する分野の分類(国際特許分類(IPC))</p> <p>Int. Cl. H03M13/22, H03M13/12</p>												
<p>B. 調査を行った分野</p> <p>調査を行った最小限資料(国際特許分類(IPC))</p> <p>Int. Cl. H03M13/00-13/22</p>												
<p>最小限資料以外の資料で調査を行った分野に含まれるもの</p> <table> <tr><td>日本国実用新案公報</td><td>1926-1996年</td></tr> <tr><td>日本国公開実用新案公報</td><td>1971-1999年</td></tr> <tr><td>日本国登録実用新案公報</td><td>1994-1999年</td></tr> <tr><td>日本国実用新案登録公報</td><td>1996-1999年</td></tr> </table>				日本国実用新案公報	1926-1996年	日本国公開実用新案公報	1971-1999年	日本国登録実用新案公報	1994-1999年	日本国実用新案登録公報	1996-1999年	
日本国実用新案公報	1926-1996年											
日本国公開実用新案公報	1971-1999年											
日本国登録実用新案公報	1994-1999年											
日本国実用新案登録公報	1996-1999年											
<p>国際調査で使用した電子データベース(データベースの名称、調査に使用した用語)</p>												
<p>C. 関連すると認められる文献</p> <table border="1"> <thead> <tr> <th>引用文献の カテゴリー</th> <th>引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示</th> <th>関連する 請求の範囲の番号</th> </tr> </thead> <tbody> <tr> <td>X Y A</td> <td>S. Dolinar and D. Divsalar, «Weight Distributions for Turbo Codes Using Random and Nonrandom Permutations», The Telecommunications and Data Acquisition Report (JPL TDA Progress Report) 42-122, Jet Propulsion Laboratory, August 15, 1995, Pasadena, California, pp. 56-65, 特にセクションIV. B. 1 (p. 61 第33-48行) 参照。</td> <td>1-4 6, 26, 27, 29 28</td> </tr> <tr> <td>X Y A</td> <td>JP.07-212250, A (富士通株式会社) 11.8月. 1995 (11.08.95) (ファミリーなし) 特に第4回参照。</td> <td>5 6, 26, 27, 29 28</td> </tr> </tbody> </table>				引用文献の カテゴリー	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号	X Y A	S. Dolinar and D. Divsalar, «Weight Distributions for Turbo Codes Using Random and Nonrandom Permutations», The Telecommunications and Data Acquisition Report (JPL TDA Progress Report) 42-122, Jet Propulsion Laboratory, August 15, 1995, Pasadena, California, pp. 56-65, 特にセクションIV. B. 1 (p. 61 第33-48行) 参照。	1-4 6, 26, 27, 29 28	X Y A	JP.07-212250, A (富士通株式会社) 11.8月. 1995 (11.08.95) (ファミリーなし) 特に第4回参照。	5 6, 26, 27, 29 28
引用文献の カテゴリー	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号										
X Y A	S. Dolinar and D. Divsalar, «Weight Distributions for Turbo Codes Using Random and Nonrandom Permutations», The Telecommunications and Data Acquisition Report (JPL TDA Progress Report) 42-122, Jet Propulsion Laboratory, August 15, 1995, Pasadena, California, pp. 56-65, 特にセクションIV. B. 1 (p. 61 第33-48行) 参照。	1-4 6, 26, 27, 29 28										
X Y A	JP.07-212250, A (富士通株式会社) 11.8月. 1995 (11.08.95) (ファミリーなし) 特に第4回参照。	5 6, 26, 27, 29 28										
<input checked="" type="checkbox"/> C欄の続きにも文献が列挙されている。		<input type="checkbox"/> パテントファミリーに関する別紙を参照。										
<p>* 引用文献のカテゴリー</p> <p>「A」特に関連のある文献ではなく、一般的技術水準を示すもの</p> <p>「E」国際出願前の出願または特許であるが、国際出願日以後に公表されたもの</p> <p>「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献(理由を付す)</p> <p>「O」口頭による開示、使用、展示等に言及する文献</p> <p>「P」国際出願日前で、かつ優先権の主張の基礎となる出願</p> <p>の日の後に公表された文献</p> <p>「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの</p> <p>「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの</p> <p>「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの</p> <p>「&amp;」同一パテントファミリー文献</p>												
国際調査を完了した日	27.01.99	国際調査報告の発送日	09.02.99									
国際調査機関の名称及びあて先 日本国特許庁 (ISA/JP) 郵便番号 100-8916 東京都千代田区霞が関三丁目4番3号		特許庁審査官(権限のある職員) 鶴葉 和生	5K 8732									
電話番号 03-3581-1101 内線 3556												

国際調査報告		国際出願番号 PCT/JP 98/05027
C(続き)		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
X	S. V. Maric, [Class of algebraically constructed permutations for use in pseudorandom interleavers], Electronics Letters, Vol. 30, No. 17, 18th August 1994, pp. 1378-1379, 特に "It is common ... permuted sequence." ( p. 1378, 右欄, 第50-53行) 参照。	7-18
A	P. Jung and M. Nasshan, [Performance evaluation of turbo codes for short frame transmission systems], Electronics Letters, Vol. 30, No. 2, 20th January 1994, pp. 111-113, 特に p. 112, 左欄, 第23-49行 参照。	21-25, 28, 30-34
X	P. Jung and M. Nasshan, [Performance evaluation of turbo codes for short frame transmission systems], Electronics Letters, Vol. 30, No. 2, 20th January 1994, pp. 111-113, 特に p. 112, 左欄, 第23-49行 参照。	19, 20
X	JP, 08-242217, A (ニイ・ティ・アンド・ティ・コーポレーション) 17.9月. 1996 (17.09.96) & EP, 715432, A2 & US, 5659580, A & CN, 1132453, A 特に段落【0019】 - 【0020】 参照。	7
X	JP, 08-97731, A (ソニー株式会社) 12.4月. 1996 (12.04.96) (ファミリーなし) 特に p. 1, 要約 参照。	7
A	JP, 07-30846, A (株式会社日立製作所) 31.1月. 1995 (31.01.95) (ファミリーなし) 特に図1, 図2 参照。	1-34
A	JP, 55-26715, A (国際電信電話株式会社) 26.2月. 1980 (26.02.80) (ファミリーなし) 特に第4図, 第5図 参照。	1-34
A	JP, 10-98397, A (株式会社富士通ゼネラル) 14.4月. 1998 (14.04.98) (ファミリーなし)	1-34
A	E. Dunscombe and F. C. Piper, [Optimal Interleaving Scheme for Convolutional Codes], Electronics Letters, Vol. 25, No. 22, 26th October 1989, pp. 1517-1518, 特に要約 参照。	1-34
P, X	渋谷 彰, 須田博人, 安達文幸, 「多重インターリーピング法のW-CDMAへの適用効果」, 電子情報通信学会技術研究報告, Vol. 97, No. 544, A-P97-178, 18.2月. 1998 (18.02.98), pp. 23-30. 特にセクション2 (p. 24, 右欄, 第1行 - p. 25, 右欄, 第24行), セクション3.3.2 (p. 27, 左欄, 第11行 - p. 27, 右欄, 第20行) 参照。	1-4
P, A		5-34

---

フロントページの続き

(81)指定国 EP(AT, BE, CH, CY,  
DE, DK, ES, FI, FR, GB, GR, IE, I  
T, LU, MC, NL, PT, SE), CA, CN, J  
P, KR, US

(注) この公表は、国際事務局（WIPO）により国際公開された公報を基に作成したものである。

なおこの公表に係る日本語特許出願（日本語実用新案登録出願）の国際公開の効果は、特許法第184条の10第1項（実用新案法第48条の13第2項）により生ずるものであり、本掲載とは関係ありません。